LNCS 3983

Marina Gavrilova et al. (Eds.)

Computational Science and Its Applications – ICCSA 2006

International Conference
Glasgow, UK, May 2006
Proceedings, Part IV

4 Part IV

Springer

Marina Gavrilova Osvaldo Gervasi
Vipin Kumar C.J. Kenneth Tan
David Taniar Antonio Laganà
Youngsong Mun Hyunseung Choo (Eds.)

Computational Science and Its Applications – ICCSA 2006

International Conference
Glasgow, UK, May 8-11, 2006
Proceedings, Part IV

Springer

Volume Editors

Marina Gavrilova
University of Calgary, Canada
E-mail: marina@cpsc.ucalgary.ca

Osvaldo Gervasi
University of Perugia, Italy
E-mail: ogervasi@computer.org

Vipin Kumar
University of Minnesota, Minneapolis, USA
E-mail: kumar@cs.umn.edu

C.J. Kenneth Tan
OptimaNumerics Ltd., Belfast, UK
E-mail: cjtan@optimanumerics.com

David Taniar
Monash University, Clayton, Australia
E-mail: david.taniar@infotech.monash.edu.au

Antonio Laganà
University of Perugia, Italy
E-mail: lag@unipg.it

Youngsong Mun
SoongSil University,Seoul, Korea
E-mail: mun@computing.soongsil.ac.kr

Hyunseung Choo
Sungkyunkwan University, Suwon, Korea
E-mail: choo@ece.skku.ac.kr

Library of Congress Control Number: 2006925086

CR Subject Classification (1998): F, D, G, H, I, J, C.2-3

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

ISSN 0302-9743
ISBN-10 3-540-34077-7 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-34077-5 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11751632 06/3142 5 4 3 2 1 0

Preface

This five-volume set was compiled following the 2006 International Conference on Computational Science and its Applications, ICCSA 2006, held in Glasgow, UK, during May 8–11, 2006. It represents the outstanding collection of almost 664 refereed papers selected from over 2,450 submissions to ICCSA 2006.

Computational science has firmly established itself as a vital part of many scientific investigations, affecting researchers and practitioners in areas ranging from applications such as aerospace and automotive, to emerging technologies such as bioinformatics and nanotechnologies, to core disciplines such as mathematics, physics, and chemistry. Due to the shear size of many challenges in computational science, the use of supercomputing, parallel processing, and sophisticated algorithms is inevitable and becomes a part of fundamental theoretical research as well as endeavors in emerging fields. Together, these far-reaching scientific areas contributed to shaping this conference in the realms of state-of-the-art computational science research and applications, encompassing the facilitating theoretical foundations and the innovative applications of such results in other areas.

The topics of the refereed papers span all the traditional as well as emerging computational science realms, and are structured according to the five major conference themes:

- Computational Methods, Algorithms and Applications
- High-Performance Technical Computing and Networks
- Advanced and Emerging Applications
- Geometric Modeling, Graphics and Visualization
- Information Systems and Information Technologies

Moreover, submissions from 31 workshops and technical sessions in areas such as information security, mobile communication, grid computing, modeling, optimization, computational geometry, virtual reality, symbolic computations, molecular structures, Web systems and intelligence, spatial analysis, bioinformatics and geocomputations, are included in this publication. The continuous support of computational science researchers has helped ICCSA to become a firmly established forum in the area of scientific computing.

We recognize the contribution of the International Steering Committee and sincerely thank the International Program Committee for their tremendous support in putting this conference together, the near 800 referees for their diligent work, and the IEE European Chapter for their generous assistance in hosting the event.

We also thank our sponsors for their continuous support without which this conference would not be possible.

Finally, we thank all authors for their submissions and all invited speakers and conference attendants for making the ICCSA Conference truly one of the premium events on the scientific community scene, facilitating exchange of ideas, fostering new collaborations, and shaping the future of computational science.

May 2006

Marina L. Gavrilova
Osvaldo Gervasi

on behalf of the co-editors
Vipin Kumar
Chih Jeng Kenneth Tan
David Taniar
Antonio Laganà
Youngsong Mun
Hyunseung Choo

Organization

ICCSA 2006 was organized by the Institute of Electrical Engineers (IEE)(UK), the University of Perugia (Italy), Calgary University (Canada) and Minnesota University (USA).

Conference Chairs

Vipin Kumar (University of Minnesota, Minneapolis, USA), Honorary Chair
Marina L. Gavrilova (University of Calgary, Calgary, Canada), Conference Co-chair, Scientific
Osvaldo Gervasi (University of Perugia, Perugia, Italy), Conference Co-chair, Program

Steering Committee

Vipin Kumar (University of Minnesota, USA)
Marina L. Gavrilova (University of Calgary, Canada)
Osvaldo Gervasi (University of Perugia, Perugia, Italy)
C. J. Kenneth Tan (OptimaNumerics, UK)
Alexander V. Bogdanov (Institute for High Performance Computing and Data Bases, Russia)
Hyunseung Choo (Sungkyunkwan University, Korea)
Andres Iglesias (University of Cantabria, Spain)
Antonio Laganà (University of Perugia, Italy)
Heow-Pueh Lee (Institute of High Performance Computing, Singapore)
Youngsong Mun (Soongsil University, Korea)
David Taniar (Monash University, Australia)

Workshop Organizers

Applied Cryptography and Information Security (ACIS 2006)

Sherman S.M. Chow (New York University, USA)
Joseph K. Liu (University of Bristol, UK)
Patrick Tsang (Dartmouth College, USA)
Duncan S Wong (City University of Hong Kong, Hong Kong)

Approaches or Methods of Security Engineering (AMSE 2006)

Haeng Kon Kim (Catholic University of Daegu, Korea)
Tai-hoon Kim (Korea Information Security Agency, Korea)

Authentication, Authorization and Accounting (AAA 2006)

Haeng Kon Kim (Catholic University of Daegu, Korea)

Computational Geometry and Applications (CGA 2006)

Marina Gavrilova (University of Calgary, Calgary, Canada)

Data Storage Devices and Systems (DSDS 2006)

Yeonseung Ryu (Myongji University, Korea)
Junho Shim (Sookmyong Womens University, Korea)
Youjip Won (Hanyang University, Korea)
Yongik Eom (Seongkyunkwan University, Korea)

Embedded System for Ubiquitous Computing (ESUC 2006)

Tei-Wei Kuo (National Taiwan University, Taiwan)
Jiman Hong (Kwangwoon University, Korea)

4th Technical Session on Computer Graphics (TSCG 2006)

Andres Iglesias (University of Cantabria, Spain)
Deok-Soo Kim (Hanyang University, Korea)

GeoComputation (GC 2006)

Yong Xue (London Metropolitan University, UK)

Image Processing and Computer Vision (IPCV 2006)

Jiawan Zhang (Tianjin University, China)

Intelligent Services and the Synchronization in Mobile Multimedia Networks (ISS 2006)

Dong Chun Lee (Howon University, Korea)
Kuinam J Kim (Kyonggi University, Korea)

Integrated Analysis and Intelligent Design Technology (IAIDT 2006)

Jae-Woo Lee (Konkuk University, Korea)

Information Systems Information Technologies (ISIT 2006)

Youngsong Mun (Soongsil University, Korea)

Information Engineering and Applications in Ubiquitous Computing Environments (IEAUCE 2006)

Sangkyun Kim (Yonsei University, Korea)
Hong Joo Lee (Dankook University, Korea)

Internet Communications Security (WICS 2006)

Sierra-Camara Josè Maria (University Carlos III of Madrid, Spain)

Mobile Communications (MC 2006)

Hyunseung Choo (Sungkyunkwan University, Korea)

Modelling Complex Systems (MCS 2006)

John Burns (Dublin University, Ireland)
Ruili Wang (Massey University, New Zealand)

Modelling of Location Management in Mobile Information Systems (MLM 2006)

Dong Chun Lee (Howon University, Korea)

Numerical Integration and Applications (NIA 2006)

Elise de Doncker (Western Michigan University, USA)

Specific Aspects of Computational Physics and Wavelet Analysis for Modelling Suddenly-Emerging Phenomena in Nonlinear Physics, and Nonlinear Applied Mathematics (PULSES 2006)

Carlo Cattani (University of Salerno, Italy)
Cristian Toma (Titu Maiorescu University, Romania)

Structures and Molecular Processes (SMP 2006)

Antonio Laganà (University of Perugia, Perugia, Italy)

Optimization: Theories and Applications (OTA 2006)

Dong-Ho Lee (Hanyang University, Korea)
Deok-Soo Kim (Hanyang University, Korea)
Ertugrul Karsak (Galatasaray University, Turkey)

Parallel and Distributed Computing (PDC 2006)

Jiawan Zhang (Tianjin University, China)

Pattern Recognition and Ubiquitous Computing (PRUC 2006)

Jinok Kim (Daegu Haany University, Korea)

Security Issues on Grid/Distributed Computing Systems (SIGDCS 2006)

Tai-Hoon Kim (Korea Information Security Agency, Korea)

Technologies and Techniques for Distributed Data Mining (TTDDM 2006)

Mark Baker (Portsmouth University, UK)
Bob Nichol (Portsmouth University, UK)

Ubiquitous Web Systems and Intelligence (UWSI 2006)

David Taniar (Monash University, Australia)
Eric Pardede (La Trobe University, Australia)

Ubiquitous Application and Security Service (UASS 2006)

Yeong-Deok Kim (Woosong University, Korea)

Visual Computing and Multimedia (VCM 2006)

Abel J. P. Gomes (University Beira Interior, Portugal)

Virtual Reality in Scientific Applications and Learning (VRSAL 2006)

Osvaldo Gervasi (University of Perugia, Italy)
Antonio Riganelli (University of Perugia, Italy)

Web-Based Learning (WBL 2006)

Woochun Jun Seoul (National University of Education, Korea)

Program Committee

Jemal Abawajy (Deakin University, Australia)
Kenny Adamson (EZ-DSP, UK)
Srinivas Aluru (Iowa State University, USA)
Mir Atiqullah (Saint Louis University, USA)
Frank Baetke (Hewlett Packard, USA)
Mark Baker (Portsmouth University, UK)
Young-Cheol Bang (Korea Polytechnic University, Korea)
David Bell (Queen's University of Belfast, UK)
Stefania Bertazzon (University of Calgary, Canada)
Sergei Bespamyatnikh (Duke University, USA)
J. A. Rod Blais (University of Calgary, Canada)
Alexander V. Bogdanov (Institute for High Performance Computing and Data Bases, Russia)
Peter Brezany (University of Vienna, Austria)
Herve Bronnimann (Polytechnic University, NY, USA)
John Brooke (University of Manchester, UK)
Martin Buecker (Aachen University, Germany)
Rajkumar Buyya (University of Melbourne, Australia)
Jose Sierra-Camara (University Carlos III of Madrid, Spain)
Shyi-Ming Chen (National Taiwan University of Science and Technology, Taiwan)
YoungSik Choi (University of Missouri, USA)
Hyunseung Choo (Sungkyunkwan University, Korea)
Bastien Chopard (University of Geneva, Switzerland)
Min Young Chung (Sungkyunkwan University, Korea)
Yiannis Cotronis (University of Athens, Greece)
Danny Crookes (Queen's University of Belfast, UK)
Jose C. Cunha (New University of Lisbon, Portugal)
Brian J. d'Auriol (University of Texas at El Paso, USA)
Alexander Degtyarev (Institute for High Performance Computing and Data Bases, Russia)
Frederic Desprez (INRIA, France)
Tom Dhaene (University of Antwerp, Belgium)
Beniamino Di Martino (Second University of Naples, Italy)
Hassan Diab (American University of Beirut, Lebanon)
Ivan Dimov (Bulgarian Academy of Sciences, Bulgaria)
Iain Duff (Rutherford Appleton Laboratory, UK and CERFACS, France)
Thom Dunning (NCSA and University of Illinois, USA)
Fabrizio Gagliardi (Microsoft, USA)
Marina L. Gavrilova (University of Calgary, Canada)
Michael Gerndt (Technical University of Munich, Germany)
Osvaldo Gervasi (University of Perugia, Italy)
Bob Gingold (Australian National University, Australia)
James Glimm (SUNY Stony Brook, USA)

Christopher Gold (Hong Kong Polytechnic University, Hong Kong)
Yuriy Gorbachev (Institute of High Performance Computing and Information Systems, Russia)
Andrzej Goscinski (Deakin University, Australia)
Jin Hai (Huazhong University of Science and Technology, China)
Ladislav Hluchy (Slovak Academy of Science, Slovakia)
Xiaohua Hu (Drexel University, USA)
Eui-Nam John Huh (Seoul Women's University, Korea)
Shen Hong (Japan Advanced Institute of Science and Technology, Japan)
Paul Hovland (Argonne National Laboratory, USA)
Andres Iglesias (University of Cantabria, Spain)
Peter K. Jimack (University of Leeds, UK)
In-Jae Jeong (Hanyang University, Korea)
Chris Johnson (University of Utah, USA)
Benjoe A. Juliano (California State University at Chico, USA)
Peter Kacsuk (MTA SZTAKI Researc Institute, Hungary)
Kyung Wo Kang (KAIST, Korea)
Carl Kesselman (USC/ Information Sciences Institute, USA)
Daniel Kidger (Quadrics , UK)
Haeng Kon Kim (Catholic University of Daegu, Korea)
Jin Suk Kim (KAIST, Korea)
Tai-Hoon Kim (Korea Information Security Agency, Korea)
Yoonhee Kim (Syracuse University, USA)
Mike Kirby (University of Utah, USA)
Dieter Kranzlmueller (Johannes Kepler University Linz, Austria)
Deok-Soo Kim (Hanyang University, Korea)
Vipin Kumar (University of Minnesota, USA)
Domenico Laforenza (Italian National Research Council, Italy)
Antonio Laganà (University of Perugia, Italy)
Joseph Landman (Scalable Informatics LLC, USA)
Francis Lau (The University of Hong Kong, Hong Kong)
Bong Hwan Lee (Texas A&M University, USA)
Dong Chun Lee (Howon University, Korea)
Dong-Ho Lee (Institute of High Performance Computing, Singapore)
Sang Yoon Lee (Georgia Institute of Technology, USA)
Tae-Jin Lee (Sungkyunkwan University, Korea)
Bogdan Lesyng (ICM Warszawa, Poland)
Zhongze Li (Chinese Academy of Sciences, China)
Laurence Liew (Scalable Systems Pte, Singapore)
David Lombard (Intel Corporation, USA)
Emilio Luque (University Autonoma of Barcelona, Spain)
Michael Mascagni (Florida State University, USA)
Graham Megson (University of Reading, UK)
John G. Michopoulos (US Naval Research Laboratory, USA)
Edward Moreno (Euripides Foundation of Marilia, Brazil)

Youngsong Mun (Soongsil University, Korea)
Jiri Nedoma (Academy of Sciences of the Czech Republic, Czech Republic)
Genri Norman (Russian Academy of Sciences, Russia)
Stephan Olariu (Old Dominion University, USA)
Salvatore Orlando (University of Venice, Italy)
Robert Panoff (Shodor Education Foundation, USA)
Marcin Paprzycki (Oklahoma State University, USA)
Gyung-Leen Park (University of Texas, USA)
Ron Perrott (Queen's University of Belfast, UK)
Dimitri Plemenos (University of Limoges, France)
Richard Ramaroson (ONERA, France)
Rosemary Renaut (Arizona State University, USA)
Reneé S. Renner (California State University at Chico, USA)
Paul Roe (Queensland University of Technology, Australia)
Alexey S. Rodionov (Russian Academy of Sciences, Russia)
Heather J. Ruskin (Dublin City University, Ireland)
Ole Saastad (Scali, Norway)
Muhammad Sarfraz (King Fahd University of Petroleum and Minerals, Saudi Arabia)
Edward Seidel (Louisiana State University, USA and Albert-Einstein-Institut, Potsdam, Germany)
Jie Shen (University of Michigan, USA)
Dale Shires (US Army Research Laboratory, USA)
Vaclav Skala (University of West Bohemia, Czech Republic)
Burton Smith (Cray, USA)
Masha Sosonkina (Ames Laboratory, USA)
Alexei Sourin (Nanyang Technological University, Singapore)
Elena Stankova (Institute for High Performance Computing and Data Bases, Russia)
Gunther Stuer (University of Antwerp, Belgium)
Kokichi Sugihara (University of Tokyo, Japan)
Boleslaw Szymanski (Rensselaer Polytechnic Institute, USA)
Ryszard Tadeusiewicz (AGH University of Science and Technology, Poland)
C.J. Kenneth Tan (OptimaNumerics, UK and Queen's University of Belfast, UK)
David Taniar (Monash University, Australia)
John Taylor (Streamline Computing, UK)
Ruppa K. Thulasiram (University of Manitoba, Canada)
Pavel Tvrdik (Czech Technical University, Czech Republic)
Putchong Uthayopas (Kasetsart University, Thailand)
Mario Valle (Swiss National Supercomputing Centre, Switzerland)
Marco Vanneschi (University of Pisa, Italy)
Piero Giorgio Verdini (University of Pisa and Istituto Nazionale di Fisica Nucleare, Italy)
Jesus Vigo-Aguiar (University of Salamanca, Spain)

Jens Volkert (University of Linz, Austria)
Koichi Wada (University of Tsukuba, Japan)
Stephen Wismath (University of Lethbridge, Canada)
Kevin Wadleigh (Hewlett Packard, USA)
Jerzy Wasniewski (Technical University of Denmark, Denmark)
Paul Watson (University of Newcastle Upon Tyne, UK)
Jan Weglarz (Poznan University of Technology, Poland)
Tim Wilkens (Advanced Micro Devices, USA)
Roman Wyrzykowski (Technical University of Czestochowa, Poland)
Jinchao Xu (Pennsylvania State University, USA)
Chee Yap (New York University, USA)
Osman Yasar (SUNY at Brockport, USA)
George Yee (National Research Council and Carleton University, Canada)
Yong Xue (Chinese Academy of Sciences, China)
Igor Zacharov (SGI Europe, Switzerland)
Xiaodong Zhang (College of William and Mary, USA)
Aledander Zhmakin (SoftImpact, Russia)
Krzysztof Zielinski (ICS UST / CYFRONET, Poland)
Albert Zomaya (University of Sydney, Australia)

Sponsoring Organizations

Institute of Electrical Engineers (IEE), UK
University of Perugia, Italy
University of Calgary, Canada
University of Minnesota, USA
Queen's University of Belfast, UK
The European Research Consortium for Informatics and Mathematics (ERCIM)
The 6th European Framework Project "Distributed European Infrastructure for Supercomputing Applications" (DEISA)
OptimaNumerics, UK
INTEL
AMD

Table of Contents

Workshop on Ubiquitous Web Systems and Intelligence (UWSI 2006)

Workshop on Ubiquitous Application and Security Service (UASS 2006)

Workshop on Embedded System for Ubiquitous Computing (ESUC 2006)

Workshop on Information Engineering and Applications in Ubiquitous Computing Environments (IEAUCE 2006)

Workshop on Component Based Software Engineering and Software Process Model (CBSE 2006)

General Tracks

Message Transport Interface for Efficient Communication Between Agent Framework and Event Service[*]

Sang Yong Park and Hee Yong Youn[**]

School of Information and Communication Engineering,
Sungkyunkwan University, 440-746, Suwon, Korea
{utri, youn}@ece.skku.ac.kr

Abstract. The multi-agent techniques have been continuously evolving as ubiquitous computing emerges as a key post-Internet paradigm. An agent dynamically executes its operations and has capabilities of self-growing and self-adaptive in open environments. Various distributed applications need to exchange asynchronous requests using an event-based execution model. To support the requests, the OMG defined a CORBA Event Service component in the CORBA Object Services (COS). Efficient interoperability between the agent framework and event service is important for achieving high performance ubiquitous applications. In this paper we propose the MTI (Message Transport Interface) for supporting such interoperability. An experiment validates the effectiveness of the proposed scheme compared to the existing omniEvent service.

Keywords: ACL, event service, MTI, multi-agent, ubiquitous computing.

1 Introduction

The multi-agent techniques have been continuously evolving as ubiquitous and pervasive computing [6] emerges as a key post-Internet paradigm. An agent dynamically executes its operations and has capabilities for autonomously solving the problems in open environment. As the technology gets mature, the key issue has become how to efficiently accomplish the task in cooperation with multi-agents in the distributed computing environments. Recently, the researchers have shown great interest in agent framework and have focused on the problems involved in the development of multi-agents and mobile agents [9-14].

One of the most popular agent frameworks is JADE [1]. It is a FIPA [2] (Foundation for Intelligent Physical Agents)-compliant software agent framework. The FIPA Agent Management specifications identify the roles of the key agents required for managing the platform, and describe the language and ontology of agent management. The mandatory roles identified for an agent platform are Agent Management System and Directory Service [8]. The FIPA-compliant agent framework basically uses the agent

[*] This research was supported in part by the Ubiquitous Autonomic Computing and Network Project, 21st Century Frontier R&D Program in Korea and the Brain Korea 21 Project in 2005.

[**] Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1 – 10, 2006.

communication language (ACL) for the communication between the agents. There exist considerable differences between the existent communication protocol and the ACL. The ACL has enough power of expression and a proper semantic structure needed for the exchange of the information among the agents. Furthermore, it guarantees autonomy which is one of the main features of the agents. In spite of these features, for the communication between the existing agent platform and non-agent platform, internal change of the agent platform or introduction of a new method is inevitable.

Various distributed applications need to exchange asynchronous requests using an event-based execution model. To support the requests, the OMG defined a CORBA Event Service [3] component in the CORBA Object Services (COS). An intermediate system for the communication between the agent platform and the non-agent platform such as event service is needed for achieving high performance ubiquitous applications. For this, we propose the MTI (Message Transport Interface) for the communication between the JADE platform and CORBA based platform supporting event service. It has been implemented in both c++ and java language. The MTI contains the monitor module reducing the message overhead and the scheduler efficiently scheduling and processing the event message extracted from the message queue. An experiment validates the effectiveness of the proposed scheme compared to the existing omniEvent service. The proposed scheme has been implemented in a middleware called CALM(Component-based Autonomic Layered Middleware)[16] developed by us.

The remainder of the paper is organized as follows. In Section 2 we describe a brief overview of JADE platform and CORBA event service. Section 3 presents the proposed scheme, and Section 4 evaluates the performance of the proposed scheme through an experiment. Finally, Section 5 concludes the paper with some remarks.

2 Background

2.1 JADE (Java Agent DEvelopment Framework)

An agent in the multi-agent system is a kind of computer program cooperating with other agents in the distributed computing environment. By collaborative operation with other agents, it can provide the users with more complicated service which is usually hard for a single application to handle. JADE is a software development framework aiming at developing multi-agent systems and applications conforming to FIPA standards for intelligent agents. It includes two main products; a FIPA-compliant agent platform and a package used to develop java agents. The main modules of it are as follows.

- AMS (Agent Management System): The AMS is the agent that exerts supervisory control over access to and use of the platform. It is responsible for maintaining a directory of resident agents and handling their life cycle.
- DF (Directory Facilitator): The DF is the agent that provides yellow page services to the agent platform. It supports a way to find an agent existing in the network.
- Main Container: Each running instance of the JADE environment is called a Container as it can contain several agents. The set of active containers is called a Platform. A single special Main Container must always be active in a platform and all other containers register to it as soon as they start.

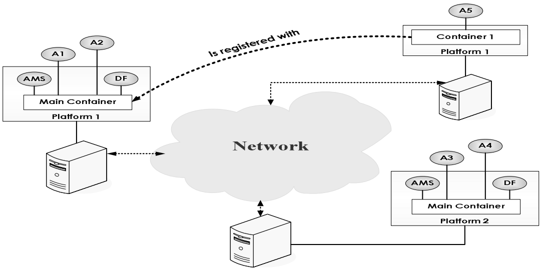
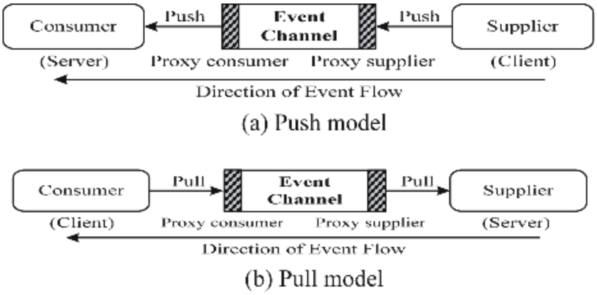


Fig. 1. An example of JADE platforms

Figure 1 illustrates an example of JADE platforms showing two JADE platforms composed by two and one container, respectively. JADE agents are identified by a unique name and, provided they know each other's name, they can communicate transparently regardless of their actual location.

2.2 CORBA Event Service

CORBA is a distributed object computing middleware specification standardized by the Object Management Group (OMG). The OMG defined a CORBA Event Service component to support asynchronous communication. Event Service allows one or more suppliers to send events to one or more consumers. Figure 2 shows the structure of COS Event Service.



(a) Push model

(b) Pull model

Fig. 2. The structure of COS Event Service

The consumers are the ultimate destinations of the events generated by the suppliers. The suppliers and consumers can both play active and passive roles, respectively. The central point of the COS Event Service is the Event Channel, which plays the role of a mediator between the consumers and suppliers. It manages object references for the supplier and consumer. It appears as a "proxy consumer" to the real suppliers on one side of the channel and a "proxy supplier" to the real consumers on the other side. The suppliers use Event Channels to push data to the consumers, and the consumers can explicitly pull data from the suppliers. The push and pull semantics of event propagation help the consumers and suppliers be free from the overly restrictive synchronous semantics of the standard CORBA two-way communication model. When developing an application using CORBA event service, he or she can employ one of the four models – push/push, push/pull, pull/push, and pull/pull – and choose a best model in terms of the design goal and characteristics of the application.

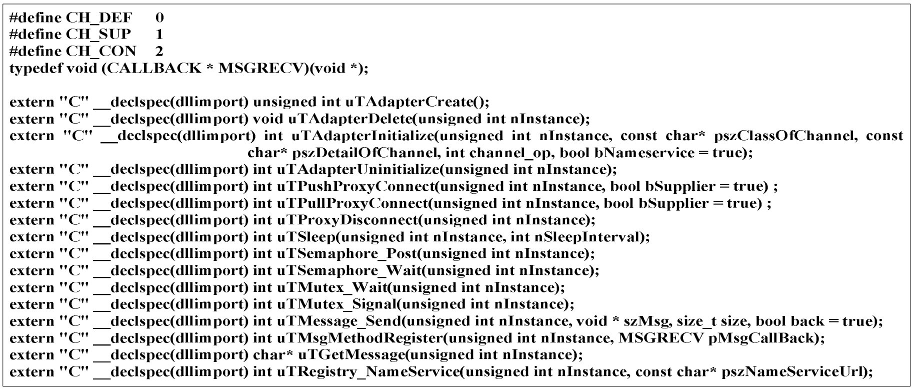
3 The Proposed Scheme

In this section we show the improved CORBA event service, and then, propose a way of implementation and design of the communication mechanism between the agent platform and CORBA event service.

3.1 Information Bus Adapter

Asynchronous communication is needed to design and program distributed applications used in ubiquitous environment. Existent event service processes event data exchanged between the supplier and consumer through statically created channels. However, the contexts in ubiquitous environments are generated in various forms according to the type and role of the sensors attached to the users and devices. Therefore, we need dynamic channel creation and management based on the context so that the events triggered by various sensors can be managed systematically and efficiently. This paper thus proposes a context-based dynamic channel creation and management scheme. The proposed event service called information bus adapter supports context-based channels between the suppliers and consumers using the channel context management module. The added module provides more sophisticated functions for channel management than the existing event service [4, 5].

The information bus adapter is developed based on omniEvents which is in charge of event service in omniORB [7]. The existing event service creates and manages static channels without considering the amount of data processed. Such manner can cause static occupation of computing resources. Therefore, it is not suitable for handling the dynamic event messages generated by the sensors and RFIDs. When the omniEvents service is in use, the information bus adapter provides the system with various interfaces as the means of converting complicate codes into a capsule like



```
#define CH_DEF     0
#define CH_SUP     1
#define CH_CON     2
typedef void (CALLBACK * MSGRECV)(void *);

extern "C" __declspec(dllimport) unsigned int uTAdapterCreate();
extern "C" __declspec(dllimport) void uTAdapterDelete(unsigned int nInstance);
extern "C" __declspec(dllimport) int uTAdapterInitialize(unsigned int nInstance, const char* pszClassOfChannel, const
                                 char* pszDetailOfChannel, int channel_op, bool bNameservice = true);
extern "C" __declspec(dllimport) int uTAdapterUninitialize(unsigned int nInstance);
extern "C" __declspec(dllimport) int uTPushProxyConnect(unsigned int nInstance, bool bSupplier = true) ;
extern "C" __declspec(dllimport) int uTPullProxyConnect(unsigned int nInstance, bool bSupplier = true) ;
extern "C" __declspec(dllimport) int uTProxyDisconnect(unsigned int nInstance);
extern "C" __declspec(dllimport) int uTSleep(unsigned int nInstance, int nSleepInterval);
extern "C" __declspec(dllimport) int uTSemaphore_Post(unsigned int nInstance);
extern "C" __declspec(dllimport) int uTSemaphore_Wait(unsigned int nInstance);
extern "C" __declspec(dllimport) int uTMutex_Wait(unsigned int nInstance);
extern "C" __declspec(dllimport) int uTMutex_Signal(unsigned int nInstance);
extern "C" __declspec(dllimport) int uTMessage_Send(unsigned int nInstance, void * szMsg, size_t size, bool back = true);
extern "C" __declspec(dllimport) int uTMsgMethodRegister(unsigned int nInstance, MSGRECV pMsgCallBack);
extern "C" __declspec(dllimport) char* uTGetMessage(unsigned int nInstance);
extern "C" __declspec(dllimport) int uTRegistry_NameService(unsigned int nInstance, const char* pszNameServiceUrl);
```

Fig. 3. The interfaces of information bus adapter

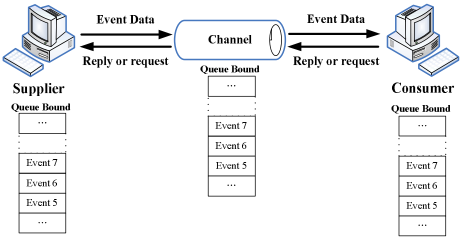


Fig. 4. Improvement of reliability of the existing CORBA Event Service

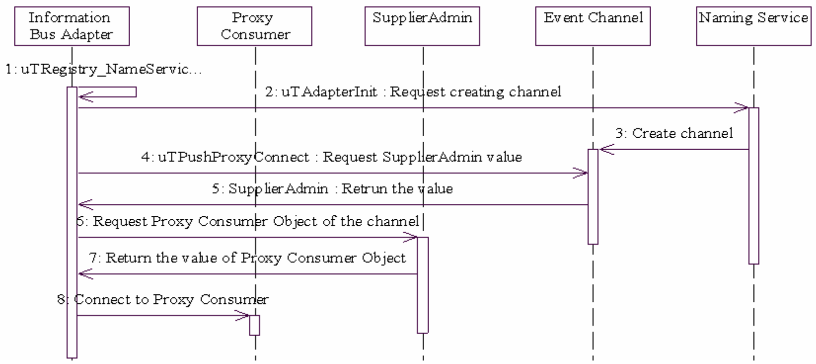

Information Bus Adapter
Proxy Consumer
SupplierAdmin
Event Channel
Naming Service
1: uTRegistry_NameServic...
2: uTAdapterInit : Request creating channel
3: Create channel
4: uTPushProxyConnect : Request SupplierAdmin value
5: SupplierAdmin : Retrun the value
6: Request Proxy Consumer Object of the channel
7: Return the value of Proxy Consumer Object
8: Connect to Proxy Consumer

Fig. 5. The sequence diagram of channel connection

Figure 3. The developers can take advantage of easy implementation; even they do not need to figure out the inner structure. In addition, this guarantees integrity of data transmission by adding new logic for upgraded reliability which the existing event service does not support yet.

The interfaces of the information bus adapter provide automatic creation/deletion/ administration of the channel and channel-name look-up service by adding the logic that can associate the name of the channel to the ORB naming service. Such functions automatically create independent channels according to the type and contents of the event context. Also, they extract data for the consumer wishing to receive a specific service. The adapter shows better performance than the case of adding the filtering logic to the consumer and supplier as in TIBCO [15]. Figure 4 shows that the proposed adapter enhances the reliability of the existing CORBA event service. Here, event channel, consumer, and supplier has an event queue storing the messages, respectively. Reliable communication is provided by letting the sender store data in a temporary storage before it sends them. Retransmission of data occurs from the storage if they are lost. The receiver transmits an ACK signal if a message is received, and then the data stored in the sender are erased. With this mechanism, the performance might decrease slightly while the reliability of the system is improved significantly.

Figure 5 shows the sequence of the operations required for a channel connection. First of all, the adapter registers the URL of the naming server using uTRegistry_NameService() interface. Then, the uTAdapterInit() interface initializes and creates a channel, and then, registers the name of the channel to the naming server. The returned object is used in the communication.

Figure 6 shows the process of passing event messages from a local object to the remote one after creating a channel. The event message is passed to the adapter using uTMessage_Send(Event) interface if an event occurs. The passed message is pushed to or pulled from the event channel by the ProxyConsumer object. If an event message arrives at the event channel, the ProxyConsumer delivers it to the ProxySupplier. The consumer wishing to receive the event passes it using the CALLBACK interface of the information bus adapter.

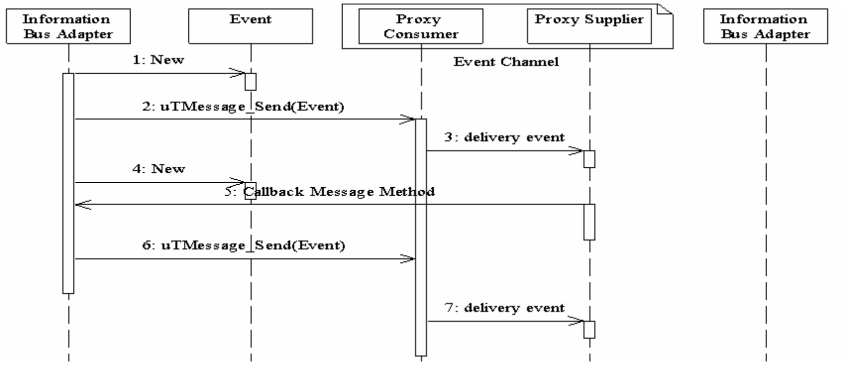


Fig. 6. The sequence diagram for sending an event message

We have developed the adapter for linux and Windows platform, including the pocket PC adapter for mobile computing. With this, we can satisfy the requirements of ubiquitous computing environments of dynamic attributes for various platforms and languages.

3.2 The Message Transport Interface

Figure 7 shows the interface used by a client for accessing the server using the interface definition language (IDL). The interface Event is defined in the IDL and it is possible to access the server from outside using the IDL. The passing_message(in EventData data) method receives event data from the adaptor, and delivers the message to the agent platform. In addition to the MTI, there exist some methods acting as the assistants that look for an agent or get the agent list. The IDL is compiled by

```
module mti {
        interface Events {
            string passing_message(in EventData data);
            string search_specified_agent(in string agent_name);
            string get_agent_list(in short list_index);
            short  get_agent_count();
             .........
             .........
            oneway void shutdown();
        };
};
```

Fig. 7. The definition of MTI

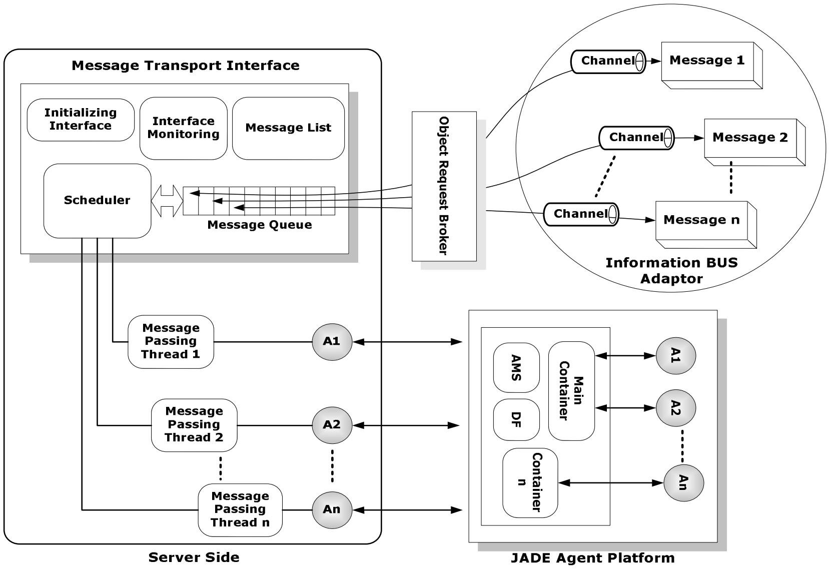


Fig. 8. The overall architecture of the system including the proposed MTI

precompiler and creates the skeleton for the implementation of server class and stub code for the client. The created code defines the implementation of server object called servant. With the technique defining the interface using the IDL, the client is able to easily connect to the server through the ORB regardless of the language used.

Figure 8 shows the overall structure of the communication between the JADE platform and non-agent platform. It consists of the information bus adapter of the client side, JADE platform, and MTI of the server side that acts as a bridge between the two platforms.

Client side: If a specific event occurs in the information bus adapter, a channel for transmitting the event message is created. The channel is classified and created automatically according to the type of the event message and contents. Also, the size of the channel is increased or decreased dynamically according to the number of events to minimize the overhead of channel administration. The object to which the generated message needs to be passed is found using the omni naming service. Next, the message is delivered to the servant object through the ORB. The adapter basically supports push or pull model.

Server side: In the server side, an instance of MTIServant class is created to initialize the ORB and MTI, and register the name context to the omniORB naming service. It also constructs MessageList to store the result of the event message at the agent platform. The MTI creates a local agent to communicate with the remote agent of JADE platform using createNewAgent() after the server is initialized. The local agent in the server side receives the agent list from the AMS agent of the remote agent platform. After that, it is able to communicate with the corresponding agent. An agent has the state of activated and suspend mode.

Agent platform: Information exchange among the agents may occur very frequently in the multi-agent system. This is controlled by the AMS (Agent Management System). An agent is executed according to the context and type of event data from the server side. After the agent platform finishes the required operation, the results are stored at arrayList in order to be returned to the client using an instance of MessageList class in the MTI.

The MTI consists of the following modules processing the event messages received from the client.

IS (Initializing System): The IS initializes the MTI for the communication between the agent platform and information bus adapter. It constructs arrayList to store the result returning from the agent platform and message queue to store the message, and activates the scheduler.
IFM (Interface Monitoring): The IFM monitors the overall MTI. It checks the overhead of message passing threads to balance the overheads.
ML (Message List): The result obtained from the cooperation among the agents in the agent platform is passed by the server again through the MTI. The ML prevents the problem occurring frequently due to data damage and improves the stability by storing the returned data at the buffer in arrayList.
MS (Message Scheduler): The MS schedules the tasks using multi-threads. When it is started, getEventMessage() is executed which brings a message from the message queue. Also, it extracts and processes the message stored in the message queue after checking the type and context of it.
MQ (Message Queue): In the MTI, there exits variable queues where messages are stored temporarily. The agent managing the MQ extracts the relevant message, grasps the type and attribute of the message, and then communicates with the remote agent.

4 Performance Evaluation

In this section the proposed scheme with the proposed MTI is evaluated by experiment, and compared with the existing omniEvent service concerning the efficiency of message transmission. The test platform includes 5 Windows XP-based PCs as the clients and 1 Solaris 8-based Sun blade 2000 as a server. The client systems used for the experiment have Intel Pentium 4 processor and 1GB main memory. In case of the proposed event service and omniEvents service 5 PCs host a supplier and the Sun blade hosts a naming server and consumer, respectively.

Figure 9 compares the process time per event. Here the Push model of Event Service is used. The proposed event service has five channels while the existing event service has five suppliers that use a default channel. Observe from the figure that process time of the proposed event service is much smaller than that of the existing event service. Notice also that, as the number of events increases, the difference gets more substantial. Observe that the average processing time of the first 10 events is slightly higher than with 20 or 30 events. This is because some delay occurs for looking up and connecting relevant agents by sending query messages to the AMS agent of the JADE platform.

Figure 10 shows that event delivery time of the proposed event service is almost same as the omniEvents service, except for relatively large size messages of 1Mb or more. This is because the proposed scheme guarantees reliable communication using buffered messages while the omniEvents does not.

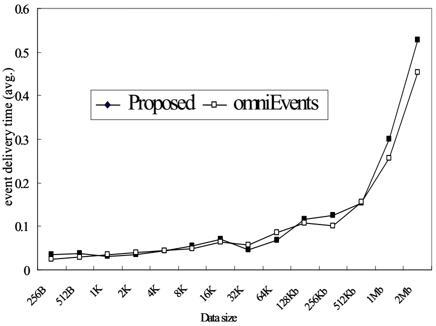


Fig. 9. The comparison of process time

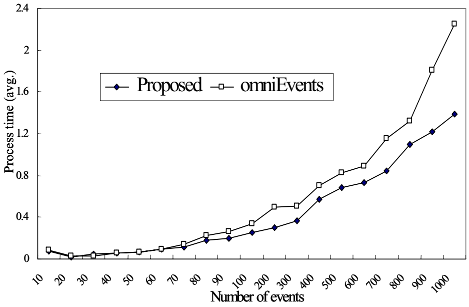


Fig. 10. The comparison of event delivery time

5 Conclusion

In this paper we have proposed a scheme which supports interoperability between the agent framework and event service without changing the internal platform. This paper has also presented an improved OMG CORBA Event Service which reduces the channel bottleneck by dynamically creating the channels and managing them based on the context referring to the events of the suppliers. The proposed event service has been implemented in a middleware called CALM(Component-based Autonomic Layered Middleware)[16] developed by us. It reduces the load of CORBA-based event servers through reflective filtering, and allows efficient event service by employing the context-based dynamic channel management. The objective of the proposed MTI is to provide a bridge between the proposed event service and agent framework. It allows transfer of the event message occurred in the information bus adapter to the JADE platform. The modules in the MTI support monitoring its interface, allow load-balancing of the messages, and improve the reliability in providing the results using the Message List. Through performance evaluation with an actual experiment, we have verified that the reliability is guaranteed and the performance is improved significantly compared to the existing omniEvents in terms of message processing time.

We need to take various characteristics into consideration along with the performance issue when designing an agent system for new applications. We will investigate the trade-offs between the performance and reliability. As the future work, on the basis of this study, we will support agent security and authorization in the proposed interface. Additionally, we will optimize the proposed scheme and develop a mechanism enhancing the QoS of the OMG CORBA Event Service. More research will also be performed on the integration of the proposed event service with other middleware such as light-weight CORBA and web service.

References

[1] JADE, Java Agent Development framework, http://jade.cselt.it.

[2] FIPA-Foundation for Intelligent Physical Agents, http://www.fipa.org.

[3] Object Management Group: Event Service Specification, Oct. 2004.

[4] H.Y. Youn et al.: Context-based Dynamic Channel Management for Efficient Event Service in Pervasive Computing, Oct. 2005, UTRI, Technical Report.

[5] S.W. Han and H.Y. Youn: A Middleware Architecture for Community Computing with Intelligent Agents, UbiCNS 2005, June 2005.

[6] M. Weiser: The Computer for the Twenty-First Century, Scientific American, Sept. 1991, pp.94-100.

[7] S.L. Lo and D. Riddoch: The omniORB version 4.0 User's Guide, AT&T Laboratories Cambridge, Oct. 2004.

[8] G. Rimassa: Runtime Supported for Distributed Multi-Agent Systems, June 2003.

[9] M. Luck, P. McBurney and C. Preist: Agent Technology: Enabling Next Generation Computing, AgentLink community. 2003.

[10] F. Bellifemine, A. Poggi, and G. Rimassa: Developing Multi-Agent Systems with a FIPA-compliant Agent Framework, Software: Practice & Experience, 2001 pp. 31: 103-128.

[11] F. Kon, R. Campbell, M.D. Mickunas, K. Nahrstedt and F.J. Ballesteros: 2K: A Distributed Operating System for Dynamic Heterogeneous Environments, in 9th IEEE International Symposium on High Performance Distributed Computing. August 1-4 2000.

[12] R. Deters: Scalability & Multi-Agent Systems, 2nd International Workshop Infrastructure for Agents, MAS and Scalable MAS., 5th Int'l conference on Autonomous Agents, May-June 2001.

[13] M. Henning and S. Vinosky: Advanced CORBA Programming with C++, addition-wesley, Boston, 1999.

[14] L.C. Lee, H.S. Nwana, D.T. Ndumu and P.De Wilde: The stability, scalability and performance of multi-agent Systems, BT Technol J. Vol. 16 No. 3 July 1998. pp.94-103.

[15] TIBCO: TIBCO Rendezvous^{TM} Concepts, Software Release 7.1, Oct. 2002, http://www.tibco.com.

[16] S.W. Han, S.K. Song and H.Y. Youn: CALM: An Intelligent Agent-based Middleware for Community Computing, 3rd Workshop on Software Technologies for Future Embedded & Ubiquitous Systems (SEUS 2006) April 2006.

An Ontology-Based Context Model in a Smart Home

Eunhoe Kim and Jaeyoung Choi

School of Computing, Soongsil University,
1-1 Sangdo-dong, Dongjak-gu, Seoul 156-743, Korea
ehkim@ss.ssu.ac.kr, choi@ssu.ac.kr

Abstract. This paper introduces an ontology-based context model in ubiquitous computing environment. We describe a Getting up scenario to show whether the model is valuable for description of context information in a home domain. We modeled context metadata as well as context information. Context metadata are defined for the use of additional context information such as probabilistic or fuzzy logic reasoning, rich understanding, or efficient context information collection. In addition we developed a context ontology server that manages and processes context information and metadata, and it is based on the ontology-based context model which is introduced in this paper. We used OWL (Web Ontology Language) for modeling the context, and extended the OWL language to represent context metadata.

1 Introduction

Ubiquitous computing environment is a computing environment which provides useful services to users by embedding computers in the physical environment and being aware of the physical circumstances. Therefore, the situational information is very important for deciding service behaviors in the ubiquitous computing environment. The situational information of entities is called context. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves [1]. A ubiquitous computing system consists of various components that acquire, interpret, process, transfer, or use the context. A context model, which is shared by the components so that they can process their tasks, should be required.

This paper suggests a context model based on ontology in smart home domains. Ontology is a formal explicit specification of a shared conceptualization, and it is applied many areas such as information retrieval, medical systems, biology information, artificial intelligent agents, e-commerce, semantic web, and so on. Ontology provides semantic interoperability between heterogeneous components in an associated smart space. The model in this paper consists of context information ontology and context metadata ontology, and uses OWL [2] which is a web ontology language. In addition a context ontology server is designed and implemented to manage and process the context ontology. The process includes reasoning and retrieval of the context ontology.

This context model describes context information as three concepts such as Entity, contextType, and Value. This approach is suitable for description of context in

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 11 – 20, 2006.

application programs and for reasoning from low-level context to high-level context. Context metadata is described using annotation properties of OWL. The annotation properties are not used in a reasoning process of OWL DL reasoner, so the properties don't affect the reasoning time and can deliver the context metadata to a context infrastructure or other applications.

This paper consists of 6 sections. Section 2 discusses the related works for this area. Section 3 explains advantages that ontology can provide when we model context. Section 4 suggests context ontology in a home domain, and shows how to model the context information and context metadata in a Getting up scenario. In this section, we also explain reasoning that uses the context model. Section 5 describes the system structure of Context ontology server, and explains the facilities of each component, and describes the implementation. Section 6 includes the conclusion and suggests future work.

2 Related Works

Recently many researches for context model in ubiquitous computing environment have been done. E-R model uses graphic notations which are extended E-R diagram to represent various characteristics such as association of context information, dependency between contexts, and quality of context [3]. E-R model and other graphical context models using UML and ORM [4,5] provide conceptual models that give understanding about context information, context characteristics, and relationships between contexts, but they lack formality.

[6] introduces an object-oriented context model for web applications in ubiquitous computing environment. This object-oriented context model is available for explicit representation of context attributes and behavior, and makes it easy to develop the ubiquitous computing applications. Both an object-oriented model and an ontology-based model define context concepts as classes, and employ the main benefits of encapsulation and inheritance. However, the objected-oriented model has a weakness that lacks logical expressiveness about context information such as symmetric, transitive, and complemented information, and so on.

COBRA-ONT [7] is an ontology-based context model which is shared by all agents of CoBrA system. COBRA-ONT models ubiquitous domain knowledge, such as people, agents, devices, events, time, and space, rather than modeling context information. CONON (OWL Encoded Context Ontology) [8] is also an ontology-based context model, but it models context information only.

In this paper, we present an ontology-based context model to overcome these shortcomings. Our model is formal and can represent rich logical characteristics of context information because it is based on ontology. Moreover, we modeled not only context information, but also context metadata for providing additional context information, rich understanding, or efficient context information collection.

3 Ontology

In general, ontology is the study or concern about what kinds of things exist in the area of philosophy. In information technology, ontology is the working model of entities and interactions in some particular domain of knowledge or practices. Ontology is, according to Tom Gruber, "the specification of conceptualizations, used to help programs and humans share knowledge" [9]. Recently, ontology-related research proceeds actively in the semantic web area to process web resources automatically using semantic information (to get machine-interpretable resource information). W3C, a web consortium organization, recommends OWL that is developed from DAML-OIL (DARPA Agent Markup Language-Ontology Inference Layer) as a language to represent ontology.

OWL is based on RDF (Resource Description Framework) that describes web resources as triples: subject, predicate, and object. OWL can define classes and properties of object, and can describe unions or intersections between classes, hierarchical relationship between properties or between classes, and constraint, complement, equivalence, differentiation, transitiveness, symmetry of classes or properties. This logical expressiveness provides the ability to deduce new information from other information which is explicitly described.

The context model based on ontology in ubiquitous computing environments has the following advantages:

- Semantic interoperability between heterogeneous hardware or software components such as devices, applications, context infrastructure, sensors, actuators by sharing common concepts of context.
- Semantic deduction facility for derived context by using ontology reasoning.
- Better understanding of the context by annotating metadata of context.

4 The Context Model

4.1 Getting Up Scenario

There are life patterns repeated everyday in human life, for example, getting up, taking a shower, eating, going out, returning home, and sleeping. Many works for ubiquitous computing applications define basic human activities in these life patterns, and support more comfortable and efficient services for basic activities. This paper introduces a Getting up scenario that helps human activities in a smart home domain. In this section, we will explain how to define and process this context information with the scenario.

Mr. Kim sets the getting up time at 6:00 am, and goes to bed late. He must go to his office early. The getting up application checks Mr. Kim's getting up time, and provides an alarm service at 6:00 am the next morning. Then the application opens the curtain to provide fresh morning air and sunshine. The application connects to a weather network service and receives local weather forecast. If it isn't rainy, the application system opens the windows. If it's rainy, the system activates the air cleaning service and then provides a light service to supply enough brightness. This getting up application checks Mr. Kim's schedule, and displays it on an output device near Mr. Kim. The system also turns on a display device to show Mr. Kim a morning TV news program. The program information is referred from the preference list which stores Mr. Kim's favorite TV program list. This service is based on follow-me service.

4.2 Modeling of Context Information

In this model, context information that describes the situation of entities consists of Entities, contextTypes, and Values. Entity represents an element of context, such as person, schedule, activity, TV, bed, and curtain. ContextType describes an attribute of the entity, such as location, power status, current activity, weather, and lighting. Value includes real data value of the contextType. For example, if we want to describe the information of a TV's power status, we can define the entity as "TV," the contextType as "powerStatus" of the TV, and the value as "ON" or "OFF" of the power status. We describe entities as owl:class constructs, and describe contextTypes as owl:datatypeProperty or owl:objectProperty constructs. In some cases, context value might contain an absolute numerical value (25.5, 0) or feature describing context (ON/OFF, Hot/Neutral/Cold). In addition, we define that 'locatedIn' and 'contains' are transitive, 'nearbyPerson' and 'atTheSameTime' are symmetric, 'locatedIn' is the inverse of 'contains', and 'uses' is the inverse of 'user'.

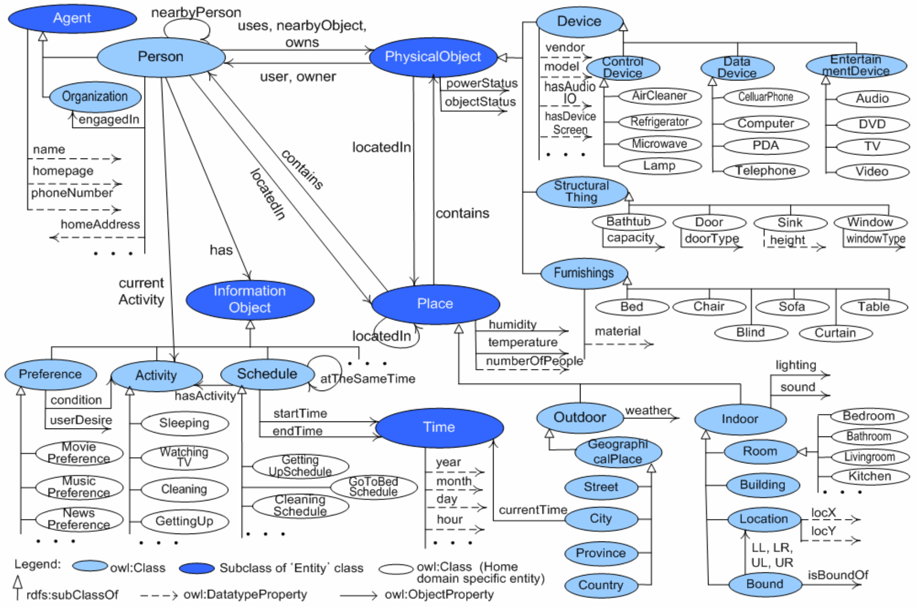


Fig. 1. Entities and ContextTypes in a Home Domain

Fig. 1 shows the structure of context information in a home domain. We define the highest level class, 'Entity', and it has subclasses (child nodes): 'Agent', 'PhysicalObject', 'InformationObject', 'Place', and 'Time'. We define 'contextType' as super property, and the other contextTypes are sub properties of this 'contextType'. These names of properties are shown on the arrow.

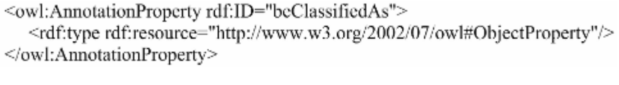
4.3 Modeling of Context Metadata

Context metadata are additional information which explains context information to improve certainty, freshness, and understanding of the context information. We suppose that all sensors can produce an error value [10]. Context information is driven from the error value; therefore the context information includes errors. This uncertain context information can affect the behavior decision of the application. Therefore, the context model needs additional elements describing confidence, accuracy, and precision of context information. Especially, ubiquitous computing environment is very dynamic and heterogeneous, so the context information is changed very rapidly, and the updating intervals also vary with context types. For freshness of context information, we need more information such as production times and average life times. We also require more information for context dependency, context sources, and context categories for better understanding of context.

Context metadata are categorized as two kinds of metadata: global and local. We divide these two categorized context metadata because of the difference of the portion where we describe the annotation. Global context metadata have values depending on contextTypes, but local context metadata have values depending on entity individuals as well as contextTypes. Therefore, we annotate global context metadata in contextType description: on the other hand, we annotate local context metadata in entity individual description. To describe context metadata, we define several owl:annotationProperties as follows. Context metadata is described using annotationProperties, and these context metadata are not interpreted nor used in OWL DL reasoning process. This information is delivered to applications or ontology servers. We will explain global and local context metadata in detail.

1) Global context metadata

beClassifiedAs. 'beClassifiedAs' annotation property type is used for classifying the context acquisition methods. There are three types of context: 'profiled', 'sensed', and 'derived'. Profiled context means data predefined by users. Sensed context acquires data from various sensors. Derived context means indirect information which is processed from other context data. Fig. 2 shows a definition part of annotation property 'beClassifiedAs', and Fig. 3 shows an example of 'beClassfiledAs' annotation in 'locatedIn' description, in which we can see that 'locatedIn' is classified as 'sensed' context.

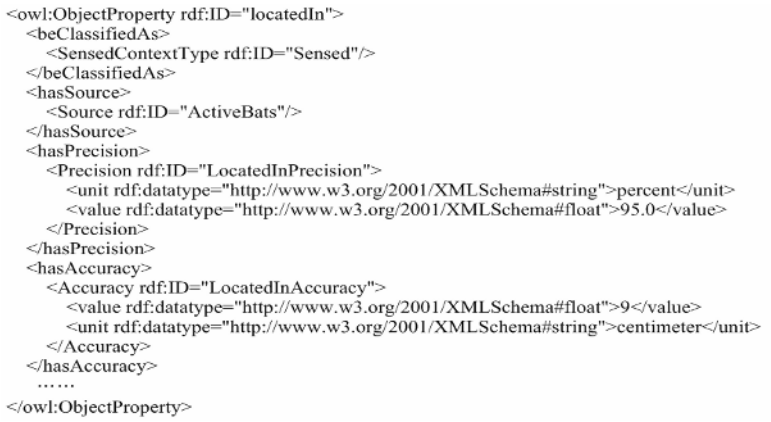


```
<owl:AnnotationProperty rdf:ID="beClassifiedAs">
   <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
</owl:AnnotationProperty>
```

Fig. 2. The description of 'beClassifiedAs' annotation property

hasSource. 'hasSource' is an annotation property in which source we can get the context data. Fig. 3 shows an example of 'hasSource' annotation in 'locatedIn' description. In this figure, the source of context data is 'ActiveBats', Active Bats Indoor Location System.

hasAccuracy / hasPrecision. 'hasAccuracy' and 'hasPrecision' are annotation properties which describe accuracy of context information. For example, we can describe the accuracy of ActiveBats as follows: "Within the given indoor area, ActiveBat is found to be accurate to within 14cm for 95% of measurement." This means that the

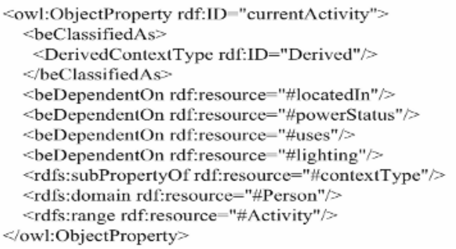


```
<owl:ObjectProperty rdf:ID="locatedIn">
  <beClassifiedAs>
    <SensedContextType rdf:ID="Sensed"/>
  </beClassifiedAs>
  <hasSource>
    <Source rdf:ID="ActiveBats"/>
  </hasSource>
  <hasPrecision>
    <Precision rdf:ID="LocatedInPrecision">
      <unit rdf:datatype="http://www.w3.org/2001/XMLSchema#string">percent</unit>
      <value rdf:datatype="http://www.w3.org/2001/XMLSchema#float">95.0</value>
    </Precision>
  </hasPrecision>
  <hasAccuracy>
    <Accuracy rdf:ID="LocatedInAccuracy">
      <value rdf:datatype="http://www.w3.org/2001/XMLSchema#float">9</value>
      <unit rdf:datatype="http://www.w3.org/2001/XMLSchema#string">centimeter</unit>
    </Accuracy>
  </hasAccuracy>
   ……
</owl:ObjectProperty>
```

Fig. 3. The global context metadata-annotated encoding for 'locatedIn'

precision of ActiveBat is 95% and the accuracy of ActiveBat is 9 centimeters. As you can see in Fig. 3, we can describe the accuracy and the precision value of 'locatedIn'; value of 'hasAccuracy' is 9 (centimeter) and value of 'hasPrecision' is 95.0 (percent).

beDependentOn. 'beDependentOn' is an annotation property for describing dependency between contexts. The 'range' of OWL annotation properties includes 'URI-reference', 'dataLiteral', and 'individual'. To add information about dependency relationships between contexts, we extend the 'range' of owl annotation properties to include a 'property' type. For this, we introduce 'owl:beDependentOn' to link a property to other properties. [11] introduced the idea of the extension. 'currentActivity' contextType, described in Fig 4, is used for description of current human activity. This type is a derived contextType from 'locatedIn', 'lighting', 'powerStatus', and 'uses' contextTypes. It means 'currentActivity' is dependent on these four contextTypes. If 'locatedIn' value (of Kim) is 'bedroom', 'uses' (of Kim) is 'bed', 'powerStatus' (of bedroom TV) is 'OFF', and 'lighting' (of bedroom bedroom) is 'Dark' or 'VeryDark', then we define Mr. Kim's 'currentActivity' is 'sleeping'.



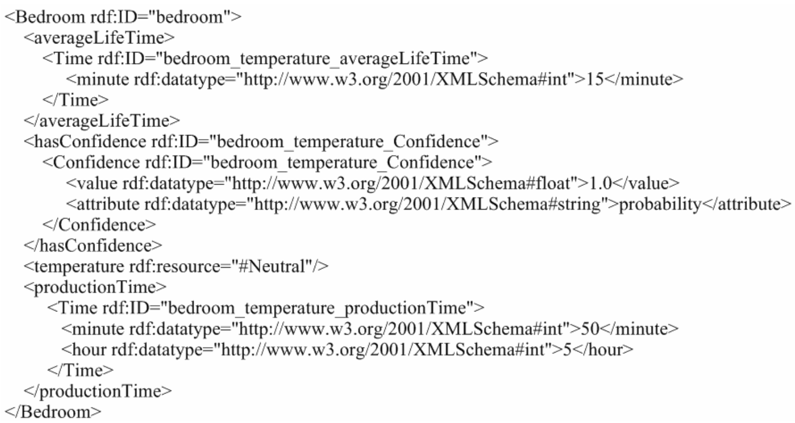
```
<owl:ObjectProperty rdf:ID="currentActivity">
  <beClassifiedAs>
   <DerivedContextType rdf:ID="Derived"/>
  </beClassifiedAs>
  <beDependentOn rdf:resource="#locatedIn"/>
  <beDependentOn rdf:resource="#powerStatus"/>
  <beDependentOn rdf:resource="#uses"/>
  <beDependentOn rdf:resource="#lighting"/>
  <rdfs:subPropertyOf rdf:resource="#contextType"/>
  <rdfs:domain rdf:resource="#Person"/>
  <rdfs:range rdf:resource="#Activity"/>
</owl:ObjectProperty>
```

Fig. 4. 'beDependentOn'-annotated encoding for 'currentActivity'

2) Local context metadata

hasConfidence. 'hasConfidence' annotation property is defined for certainty of context information. The certainty value measures probability (in case of probabilistic measurement approach) or membership value (in case of fuzzy logic measurement approach), and its value is usually between 0 and 1. Fig. 5 shows the description of ‘hasConfidence’. In the description, we can see the confidence value is 1.0 and it uses probability measurement approach.

productionTime. ‘productionTime’ is an annotation property for metadata description of context information creation time. In Fig. 5, the information creation time of bedroom lighting is 5:50 a.m.



```
<Bedroom rdf:ID="bedroom">
  <averageLifeTime>
    <Time rdf:ID="bedroom_temperature_averageLifeTime">
       <minute rdf:datatype="http://www.w3.org/2001/XMLSchema#int">15</minute>
    </Time>
  </averageLifeTime>
  <hasConfidence rdf:ID="bedroom_temperature_Confidence">
    <Confidence rdf:ID="bedroom_temperature_Confidence">
       <value rdf:datatype="http://www.w3.org/2001/XMLSchema#float">1.0</value>
       <attribute rdf:datatype="http://www.w3.org/2001/XMLSchema#string">probability</attribute>
    </Confidence>
  </hasConfidence>
  <temperature rdf:resource="#Neutral"/>
  <productionTime>
     <Time rdf:ID="bedroom_temperature_productionTime">
       <minute rdf:datatype="http://www.w3.org/2001/XMLSchema#int">50</minute>
       <hour rdf:datatype="http://www.w3.org/2001/XMLSchema#int">5</hour>
     </Time>
  </productionTime>
</Bedroom>
```

Fig. 5. The local context metadata-annotated encoding for ‘bedroom’

averageLifeTime. ‘averageLifeTime’ is an annotation property to describe an average interval of valid information updating. For example, ‘averageLifeTime’ value of bedroom lighting is 15 minutes in Fig. 5.

4.4 Context Reasoning

Logical characteristics of contextTypes and logical relationship information between contextTypes deduce new context information through an ontology reasoner. We use a simple ontology description method, a triple of (Entity, contextType, Value), for simple explanation of context reasoning. We defined ‘locatedIn’ contextType as a transitivity property. If property P is tagged as transitivity, then for any x, y, and z: (x,P,y)∧(y,P,z) iff (x,P,z). For example, context information (windows1, locatedIn, bedroom) and (bedroom, locatedIn, KimsHouse) can deduce other information (windows1, locatedIn, KimsHouse) using an ontology reasoner. Transitivity characteristic properties, symmetric and inverseOf peroperties explained in Section 4.2 are used.

We used a rule-based inference reasoner to produce derived context from context ontologies. As we explained in the ‘beDependentOn’ part of subsection 4.3, dependency information of the derived context is described as ‘beDependentOn’ annotation property. The rule to deduce (Kim, currentActivity, Sleeping) is as follows:

(Kim, locatedIn, bedroom) ∧ (Kim, uses, bed)
∧ (bedroom, lighting, Dark or VeryDark) ∧ (bedroomTV, powerStatus, OFF)
→(Kim, currentActivity, sleeping)

'beDependentOn' metadata, described in the derived contextType, provides a hint for dependency between contexts to users who define rules. Besides the metadata, 'hasConfidence' metadata of context can be used for reasoners which use probabilistic or fuzzy logic mechanism.

5 Context Ontology Server

In this paper, we design and develop a context ontology server in a smart home domain to manage and process context ontology. The context ontology server consists of Context ontology processor, Context query processor, and Context retrieval commandline interface. Fig. 6 shows the structure in detail. Fig. 6 consists of three parts. The top part describes Getting up scenario diagram, lower-left part shows Scenario editor, and lower-right part is the layout of Context ontology server.

Context ontology processor consists of two modules: Context ontology reasoning module and Rule-based context reasoning module. Context ontology reasoning module reasons using context schema and current context information. Context schema is an XML representation of context model which is suggested in Section 3. Context aggregator collects current context information from various sensors in a home domain. Rule-based context reasoning module has facilities to deduce derived contexts. The processor also uses productionTime and averageLifeTime context metadata for updating context ontology.

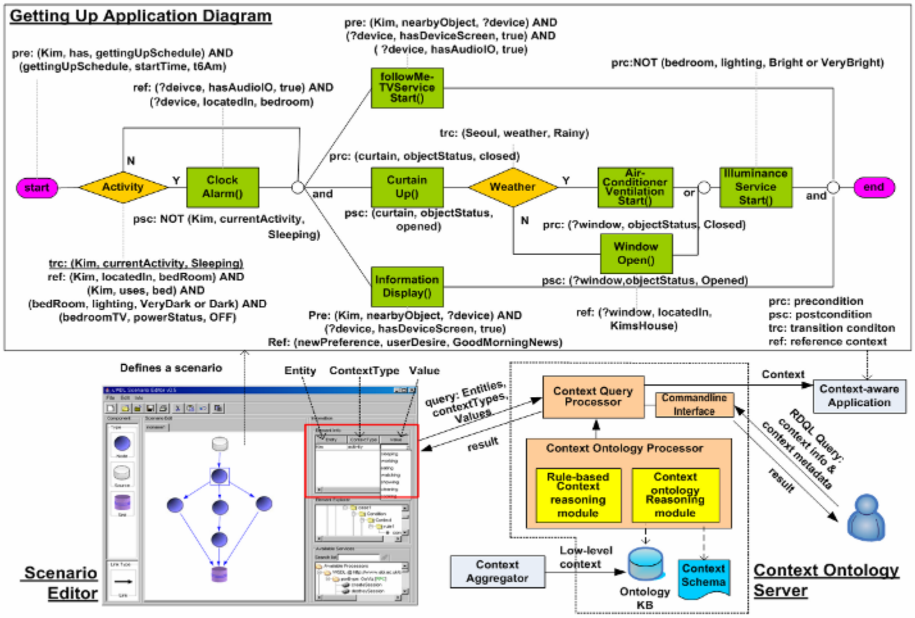

Getting Up Application Diagram
pre: (Kim, has, gettingUpSchedule) AND (gettingUpSchedule, startTime, t6Am)
ref: (?deivce, hasAudioIO, true) AND (?device, locatedIn, bedroom)
pre: (Kim, nearbyObject, ?device) AND (?device, hasDeviceScreen, true) AND (?device, hasAudioIO, true)
prc:NOT (bedroom, lighting, Bright or VeryBright)
followMe-TVService Start()
trc: (Seoul, weather, Rainy)
prc: (curtain, objectStatus, closed)
N
start
Activity
Y
Clock Alarm()
and
Curtain Up()
Weather
Y
Air-Conditioner Ventilation Start()
or
Illuminance Service Start()
and
end
psc: NOT (Kim, currentActivity, Sleeping)
psc: (curtain, objectStatus, opened)
N
prc: (?window, objectStatus, Closed)
Window Open()
trc: (Kim, currentActivity, Sleeping)
ref: (Kim, locatedIn, bedRoom) AND (Kim, uses, bed) AND (bedRoom, lighting, VeryDark or Dark) AND (bedroomTV, powerStatus, OFF)
Information Display()
psc: (?window,objectStatus, Opened)
Pre: (Kim, nearbyObject, ?device) AND (?device, hasDeviceScreen, true)
Ref: (newPreference, userDesire, GoodMorningNews)
ref: (?window, locatedIn, KimsHouse)
prc: precondition
psc: postcondition
trc: transition conditon
ref: reference context
Defines a scenario
Entity
ContextType
Value
query: Entities, contextTypes, Values
result
Scenario Editor
Context Query Processor
Commandline Interface
Context
Context-aware Application
Context Ontology Processor
Rule-based Context reasoning module
Context ontology Reasoning module
RDQL Query: context info & context metadata
result
Low-level context
Context Aggregator
Ontology KB
Context Schema
Context Ontology Server

Fig. 6. Context ontology Server, Scenario editor, and Getting up application diagram

Context query processor is a query engine and provides context retrieval services requested by applications and a scenario editor. Context retrieval commandline interface is a Windows-based interface which can be used to search context information or metadata. To search for context information or metadata, the commandline interface provides users with RDQL which is a query language for RDF.

Scenario editor is a tool for describing the flow of services in ubiquitous applications. The execution, termination, and transition of services depend on the context information. Therefore, Scenario editor requests available context from Context query processor for describing pre-condition, post-condition, and transition condition of the services. Getting up application diagram shows the flows of services in the application which is produced by scenario editor, and it also shows context information related to the services. Our Context ontology server is implemented by using Jena 2.2 ontology API which is developed by HP. We implemented the Rule-based context reasoning module using GeneralRuleReasoner which includes RETE engine or one tabled engine supported by Jena, and Context ontology reasoning module using Racer which is a well-known OWL DL reasoner.

6 Conclusion

This paper suggests an ontology-based context model in a home domain. This model has an advantage in that it can represent context metadata as well as context information. By using context metadata described with OWL annotation properties; this context model can provide additional information for applications, reasoners, or context infrastructure which need context metadata for probabilistic or fuzzy logic reasoning, rich understanding, or efficient context information collection. Using annotation properties doesn't affect the reasoning process of OWL DL reasoner, so it doesn't increase the reasoning time.

This context model is different from CONON or COBRA-ONT because it uses metadata. These two models use only context information. E-R Model also provides metadata modeling method like our model, but it is not formal because it uses graphic notations. Our model is formal because it uses ontology language OWL. [6] uses an object-oriented context modeling method. The object-oriented method makes it easy to develop ubiquitous computing applications, but as we mentioned in Section 2 it has a limitation of representation of context's logical characteristics. On the other hand, our model is available to represent the logical characteristics of context information.

Our model has a limitation because our model extends the range of OWL annotation properties to describe dependency between contexts. It causes decreasing compatibility between OWL languages, but it is not a serious problem because annotation properties are only used during the annotation process.

As a related work, we are developing a context infrastructure. It uses the Context ontology model and the Context ontology server which are introduced in this paper. In the near future, we will develop an integrated model that includes a sensor, actuator, and service model. These ontology-based sensor, actuator, and service model are strongly related to our context model in this paper, so context metadata or context information can be used for a variety of purposes.

Acknowledgement. This work was supported by Korea Research Foundation Grant (KRF-2004-005-D00172).

Reference

1. Anind K. Dey and Gregory D. Abowd: Towards a Better Understanding of Context and Context-Awareness. Workshop on The What, Who, Where, When, and How of Context-Awareness (CHI 2000), The Hague, The Netherlands, April 3, (2000)
2. OWL: http://www.w3.org/2004/OWL/
3. Karen Henricksen et al.: Modeling Context Information in Pervasive Computing Systems, Pervasive 2002, LNCS 2412, pp. 167-180, (2002)
4. Joseph Bauer: Identification and Modeling of Contexts for Different Information Scenarios in Air Traffic, Diplomarbeit, March. (2003)
5. Karen Henricksen, et al.: Generating Context Management Infrastructure from High-Level Context Models, MDM 2003, January, (2003)
6. G. KAPPEL et al.: Customisation for Ubiquitous Web Applications - A Comparison of Approaches, International Journal of Web Engineering and Technology (IJWET), Inaugural Volume, No 1, Inderscience Publishers, (2003)
7. COBRA-ONT, http://cobra.umbc.edu/ontologies.html
8. Xiaohang Wang, et al.: Ontology-Based Context Modeling and Reasoning using OWL, Workshop on Context Modeling and Reasoning at PerCom'04, Orlando, Florida, (2004)
9. T. Gruber, A Translation Approach to Portable Ontology Specification, in Knowledge Acquisition Journal, Vol. 5, pp. 199-220, (1993)
10. Jeffrey Hightower et al.: A Survey and Taxnomy of Location Systems for Ubiqutious Computing, Technical Report UW-CSE 01-08-03, University of Washinton, August, (2001)
11. Yi Yang and Jacques Calmet: OntoBayes: An Ontology-Driven Uncertainty Model, IAWTIC, Vienna, November, (2005)

Service Mobility Manager for OSGi Framework

Seungkeun Lee[1], Intae Kim[1], Kiwook Rim[2], and Jeonghyun Lee[1]

[1] Department of Computer Science and Engineering,
Inha University, Inchon, Korea
{sglee, inking}@nlsun.inha.ac.kr, jhlee@inha.ac.kr
[2] Department of Computer and Information Science,
Sunmoon University, Asan, Choongnam, Korea
rim@sunmoon.ac.kr

Abstract. The Open Services Gateway Initiative (OSGi) attempts to meet the ubiquitous computing environment by providing a managed, extensible framework to connect various devices in a local network such as in a home, office, or automobile. By defining a standard execution environment and service interface, the OSGi promotes the dynamic discovery and collaboration of devices and services from different sources. The OSGi offers a unique opportunity for ubiquitous computing as a potential framework for achieving interoperability between various sensors, home appliances, and networked devices. The OSGi framework supports a remote installation of a bundle, which is a unit that installs and deploys services. However, in order for the service in execution to move, a specific form of bundle such a mobile service manager is needed, one which is able to move through a heterogeneous network.This paper proposes a method that can manage bundles for supporting dynamic service's mobility between frameworks, in order to ensure the mobility of services in a multiple the OSGi framework environment. For our purposes, we have designed the mobile service management system for managing the lifecycle of the bundle and for the mobility of services in the OSGi framework. The mobile service management system we are proposing implements a bundle form which can perform in an OSGi framework as well as manage the mobile services. As a result, mobility in a ubiquitous computing environment will be supported more efficiently.

1 Introduction

OSGi is an industry plan regarding the standards for allowing sensors, embedded computing devices and electronic appliances to access the Internet, and it provides a Java-based open standard programming interface for enabling communication and control between service providers and devices within home and small business networks[1,2]. Whereas previous technologies focused on interoperation among devices, OSGi places emphases on service delivery, allocation and management for the devices. Furthermore, linkage services concerning Jini or UPnP can be deployed or interacted based on development of OSGi-based applications. OSGi is already providing connections for various devices in local networks such as the home, office and vehicle and providing manageable and expandable frameworks, expediting the arrival of the ubiquitous computing environment. Moreover, by defining standard execution

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 21 – 29, 2006.

environments and service interfaces, OSGi enhances dynamic findings and collaboration among various heterogeneous resources[3][4].

An OSGi-based system has a structure for distributing new services, and the structure consists of only the elements on the local network, giving it relatively closed characteristics. However, while service management and distribution can be dynamically executed within such single OSGi framework, there is insufficient support for applications with mobility among multiple frameworks. Therefore, there must be sufficient consideration for mobility of the users, devices and sensors among multiple OSGi frameworks in the expanded smart space, calling for research efforts in services supporting such mobility. Since the user, devices and sensors have mobility in a smart space, the services need to be mobile with their execution statuses stored. Also, mobility management must be efficiently supported for such service elements in the expanded smart space where multi-dimensional OSGi frameworks are located.

For example, let's say a user is heading toward home while listening to an mp3 music files on the PDA. Upon arrival at home, if the user wishes to listen to the same music on the PC, there is a cumbersome task of having to select the music play list from the PC directory. Even if there is an identical music play list in the PC, which eliminates the need for the user to select each individual song, there would not be the satisfaction of continuously listening to the music that had been playing on the PDA. However, if the mp3 player can be moved from the PDA to the PC, the music play list and song information can be maintained, allowing the user to appreciate the music without interruption[5][6][7].

This study deals with designing an OSGi-based framework using a mobile agent technology that supports mobility and duplication with status information in the distribution environment. By supporting bundles in the form of a mobile agent, the designed framework also supports mobility of the bundles within multiple OSGi system environments. Therefore, it can support mobility of various elements such as services for specific components or users as well as device drivers. In order to do so, the OSGi framework's open source Knopflerfish 1.3.3 is expanded and a mobile agent management system is designed to support the bundles' mobile lifecycle[8][9].

2 OSGi (Open Services Gateway Initiative)

OSGi is a non-profit organization that defines standard specifications for delivering, allocating and managing services in the network environment. In its initial stage, OSGi was focused on the home service gateway, but it has recently expanded from a specific network environment to the ubiquitous environment. In turn, the objective of OSGi has become implementation of the service gateway for diverse embedded devices and their users[10].

The OSGi service platform displayed in Fig. 1 consists of the OSGi framework and standard services. The OSGi framework signifies the execution environment for such services and includes the minimum component model, management services for the components and the service registry. A service is an object registered in a framework used by other applications. For some services, functionality is defined by the interfaces they implement, allowing different applications to implement identical "service" types.

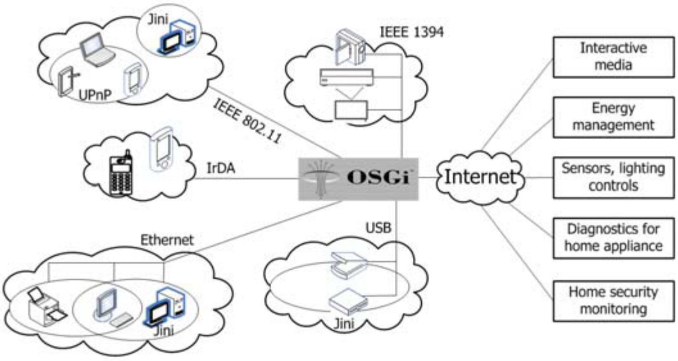


Fig. 1. The Overview of OSGi

The OSGi framework installs an OSGi component called a "bundle" and supports a programming model for service registration and execution. Furthermore, the framework itself is expressed with a bundle, which is referred to as a "system bundle". The roles of the OSGi framework that provides the hosting environment for the bundle can be summarized as follows.

- Bundle lifecycle management
- Resolution of independence among bundles
- Registry management for services
- Event processing for bundle status modification, service registration/removal and framework actions

Bundles are service groups using services registered in the service registry as well as component units. Service implementation is delivered to and allocated in the framework through a bundle, which is a physical and logical unit. From the physical perspective, a bundle is distributed in a Java archive file format (JAR) that includes codes, resources and manifest files. The manifest file informs the framework of the bundle class execution path and declares Java packages that will be shared with other bundles. It also has information regarding the bundle's activator class. The concept of a single bundle is similar to that of a single process in an operating system. From the logical perspective, it is a service provider for a certain service or a service requester that intends to use a particular service within the framework during the execution time. The OSGi framework provides the mechanism for managing the bundle lifecycle. If a bundle is installed and executed in a framework, it can provide its services. Moreover, it can find and use other services available on the framework and can be grouped with other bundle services through the framework's service registry. When registering a service in the OSGi framework's service registry, a bundle stores the service attributes in the form of the attribute-value pair. Such service attributes are used so that they can be distinguished among multiple service providers.

Once installed and started, a bundle distributed as a JAR file is operated by the Activator class. The bundle lifecycle is depicted in Fig. 2. In most cases, it is directly affected by the framework's management mechanism. After installation and prior to

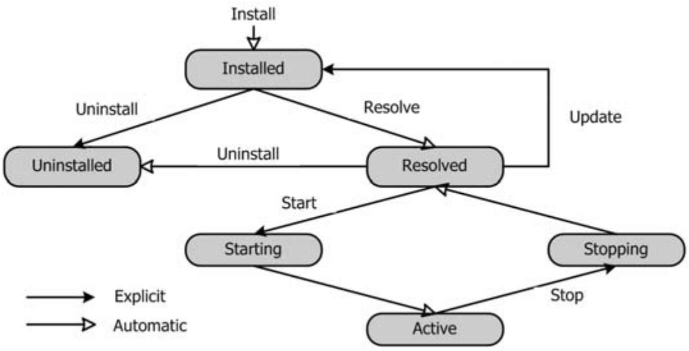


Fig. 2. The lifecycle of the bundle

execution, a bundle must be prepared in the resolved state. The framework checks dependence on external Java packages to place the bundle in the resolved state. This is for the framework to check whether the requested bundle can be used or approached in case there is any dependence present. A bundle carries out a particular task designed by the developer by implementing a BundleActivator interface of the Activator class. The Activator class called by the framework is implemented along with start and stop methods. Once called, such methods obtains the execution environment (bundle context) that allows framework functions such as access to other bundles, bundle management task execution, service registration, search for other services and listener registration for different events, enabling the bundle to indirectly access the OSGi framework functions[11][12].

3 System Design

Service Mobility Manager consists of Mobility Interface, Service Serializer, SOAP Manager. Fig. 3. describes Service Mobility Manager. The overall structure consists of the mobile interface for managing mobility, elements for processing serialization/deserialization and elements for SOAP message transmission/reception. When the Mobility Interface receives a mobility request from a bundle service, it manages the service bundle's lifecycle. The status information prior to mobility is marshalled into XML by the ServiceSerializer, and the SOAPClient delivers the SOAP message to the destination. The class file is installed through the destination BundleInstaller, and the ServiceDeserializer resumes the service by unmarshalling the SOAP message into an object.

3.1 The Extended Lifecycle for Service Bundle

In this paper, we extend the status of bundle in OSGi which compose of 'Resolved', 'Starting', 'Active' and 'Stopping', adding 'DEAD', 'Movable', 'Moved'. 'Dead' status is different from 'Unistalled' which means that bundles are omitted automatically. 'DEAD' is used for checking information of service before it is received move

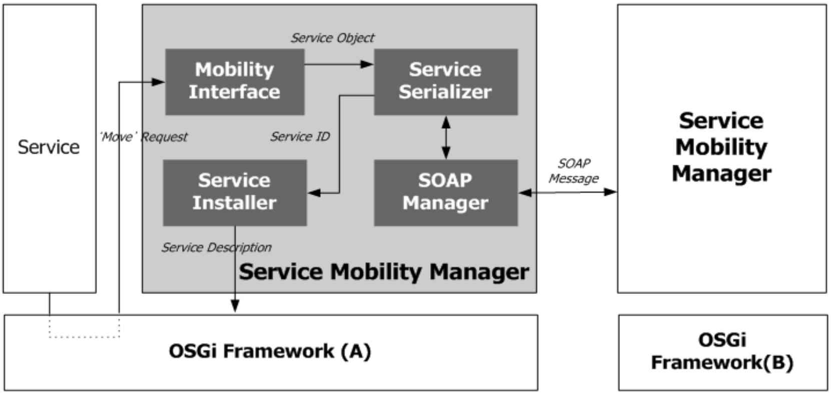


Fig. 3. The structure of the Service Mobility Manager

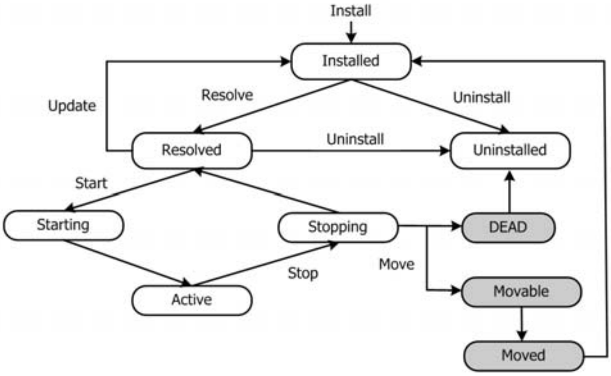


Fig. 4. The extended lifecycle for service bundle

request. The status of bundle is changed 'Movable' to 'Moved' for the service which is received move request. 'Move' status is used for the process which OSGi framework sends service to other OSGi framework. If OSGi framework would finish sending of service to other OSGi framework, service status is changed to 'Uninstalled', and this bundle is removed.

3.2 Mobility of Service

Upon receipt of a mobility request, the service is switched to the mobility request state and execution suspension is requested. The service receiving the request returns after completing the action currently in process. If converted to the mobile state, the status information is serialized into the XML format using the Serializer, and service mobility is requested to the SOAPClient. The SOAPClient verifies the destination and generates an SOAP message to call SOAPService to the destination. If mobility is successful, the service currently being executed is deleted from the registry.

At the destination, the SOAPService waiting for the SOAP message receives the URL information regarding the class location as well as the serialized data and delivers them to the MobileBundleManager. Prior to deserializing the received object, the MobileBundleManager installs the bundle from a remote location through the BundleInstaller. Upon successful installation, the object is deserialized and restored to the state prior to mobility. Finally, the service is converted to the RUNNABLE state and registered at the service registry. Algorithm 1 describes the process of service transmission between OSGi frameworks.

Algorithm 1. Sending ServiceObject with ServiceID

```
Input
ServiceID : ServiceID registered in Service Registry-
Variables
ServiceRef :Service Reference
ServiceDes :Service Description
ServiceStatus :  Service Status Information
SOAPMessage :  SOAPMessage used in Service Transmission
Begin Algorithm
  ServiceRef =
  ServiceManager.ServiceFinder.GetServiceRef(ServiceID)
  ServiceDes =
  ServiceManager.ServiceFinder.GetServiceDes(ServiceID)
  res = ServiceRef.beforeMoving()
  If (res is true)
  Begin If
    ServiceStatus = ServiceSerializer.serializer(res)
    SOAPMessage = SOAPService.makeSOAPMessage(URL,
                  ServiceStatus, ServiceDes)
    sendMessage(TargetURL, SOAPMessage)
  End If
End Algorithm
```

4 System Implementation

Table 1. displays the overall software configuration used for implementing the OSGi-based mobile agent management system proposed in this paper.

During SOAP transmission from Knopflerfish using the Apache Axis, a minor bug was found in Java 1.5 or later, so the version 1.4.2 was used.

Table 1. The software configuration

Software configuration	Description
SOAP	Apache Axis 1.2
Java Binding	Castor 0.97
OSGi Framework	Knopflerfish 1.3.3
Java Virtual Machine	Java 1.4.2
Operating System	Microsoft Windows XP

When moving an object, it was sent to the SOAP Body element was a parameter by serializing it into an XML format as below. Castor and Apache Axis were used for serialization and SOAP transmission, respectively. Other resource and class files were not sent as attached files to the SOAP message. Rather, a mobile agent bundle was installed using the bundle allocation function at the remote location.

The example of the trasferring SOAP message

```
<soapenv:Envelope ....>
  <soapenv:Body>
    <ns1:service xmlns:ns1="http://webserivce.ema">
      <obj xsi:type="xsd:string">
      &lt;agent&gt;
        &lt;status&gt;5&lt;/status&gt;
        &lt;action&gt;
          &lt;name&gt;Simple Action&lt;/name&gt;
          &lt;msg&gt;goodluck&lt;/msg&gt;
        &lt;/action&gt;
      &lt;/agent&gt;</obj>
    </ns1:service>
  </soapenv:Body>
</soapenv:Envelope>
```

The procedure and configuration for creating mobile bundles using the framework implemented in this paper are as follows.

First, in order to implement a user-defined agent class, an XML schema must be created afterwards for object serialization using Castor in a format where the agent class defined in the agent bundle is inherited.

The Manifest file of the user-defined agent designated the Bundle-Activator as the MobileBundleActivator, for which the ema.core.activator package in the agent bundle was imported. The MobileBundleActivator class reads the Agent-Class and Agent-Name header values indicated in the Manifest using the bundle objective and registers them in EAR. In order to register a user agent as a service from the AgentManager bundle, the agent class registered in EAR must be dynamically loaded. For this purpose, the user-defined agent was limited to the agent.impl package.

The example of the Manifest file

```
Bundle-Activator:ema.core.activator.
         MobileBundleActivator
Agent-Name: MyAgent
Agent-Class: agent.impl.MyAgent
Import-Package: org.osgi.framework,
         ema.core.activator
...
```

5 Experiment

In order to test the service mobile framework proposed in this paper, an MPlayer bundle was developed for playing MP3 music files. JVM and OSGi mobile agent framework bundles were installed in a PDA and PC, respectively as an experiment environment as shown in Fig 5.

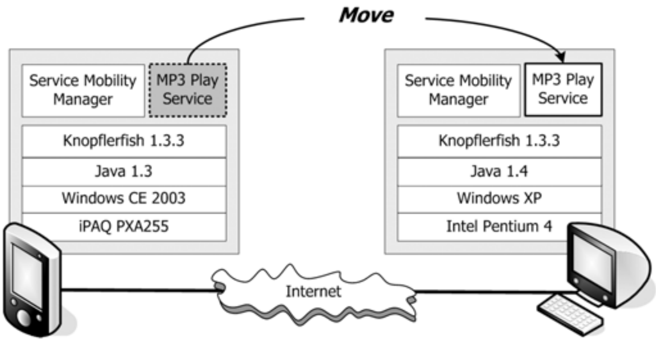
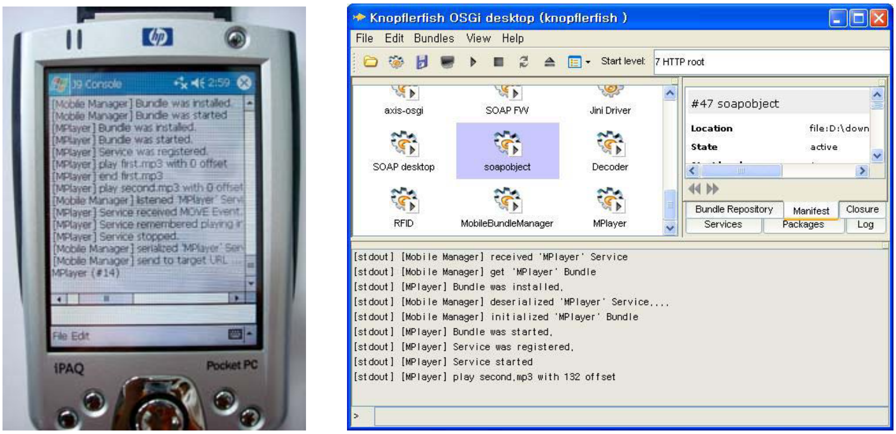


Fig. 5. The mobility of the MPlayer bundle

When a user listening to music from the MPlayer installed in the PDA moves into the space where the PC is located, the MPlayer service was moved to the PC and the file was resumed from the point where it had been playing from the PDA, providing a continuous MPlayer service to the user.

Fig. 6 (a) displays the MPlayer in use from the PDA prior to service mobility. The state data serialized during mobility is the offset information regarding the music play list and the music file currently being played. The MPlayer does not have a GUI, and it is a bundle that plays the mp3 play list through a simple configuration file.

Fig. 6 (b) is the result screen after bundle mobility. The MPlayer bundle is automatically downloaded and installed using the bundle's class loading function, the service is initialized with the music play list offset data, and the music is played. Implementation of the prototype displayed that the OSGi-based mobile agent system proposed by this paper can operate as intended without much problem.



```
[Mobile Manager] Bundle was installed.
[Mobile Manager] Bundle was started
[MPlayer] Bundle was installed.
[MPlayer] Bundle was started.
[MPlayer] Service was registered.
[MPlayer] play first.mp3 with 0 offset
[MPlayer] end first.mp3
[MPlayer] play second.mp3 with 0 offset
[Mobile Manager] listened 'MPlayer' Servi
[MPlayer] Service received MOVE Event.
[MPlayer] Service remembered playing ir
[MPlayer] Service stopped.
[Mobile Manager] serialized 'MPlayer' Ser
[Mobile Manager] send to target URL ...
MPlayer (#14)
```

```
[stdout] [Mobile Manager] received 'MPlayer' Service
[stdout] [Mobile Manager] get 'MPlayer' Bundle
[stdout] [MPlayer] Bundle was installed.
[stdout] [Mobile Manager] deserialized 'MPlayer' Service....
[stdout] [Mobile Manager] initialized 'MPlayer' Bundle
[stdout] [MPlayer] Bundle was started.
[stdout] [MPlayer] Service was registered.
[stdout] [MPlayer] Service started
[stdout] [MPlayer] play second.mp3 with 132 offset
>
```

(a) Before moving of service

(b) After moving of service

Fig. 6. The mobility of the MPlayer service

6 Conclusion

Service mobility manager is inevitable in order to provide object mobility among OSGi frameworks constituting the ubiquitous computing environment such as the home network. This paper proposed a bundle in the form of a mobile service that can be autonomously executed in the OSGi framework, for which a mobile service lifecycle and a service mobility management system were designed and implemented for managing mobility. The designed mobile agent management system was implemented in a bundle format to operate in the OSGi framework, and it also allowed dynamic management of autonomous services to provide mobility in a more efficient manner.

In order to provide intelligent services in the future, there should be research efforts regarding OSGi-based situation recognition frameworks using the mobile service technology as well as security considerations for the mobile agent.

Acknowledgement

This research was supported by the MIC(Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Assessment) (IITA-2005-C1090-0502-0031).

References

1. Open Services Gateway Initiative. http://www.osgi.org
2. D. Marples and P. Kriens, “The Open Services Gateway Initiative: An Introductory Overview,” *IEEE Communications Magazine*, Vol. 39, No. 12, pp.110-114, December 2001
3. C. Lee, D. Nordstedt, and S. Helal, “Enabling Smart Spaces with OSGi,” *IEEE Pervasive Computing*, Vol. 2, Issue 3, pp.89-94, July_Sept. 2003
4. P. Dobrev, D. Famolari, C. Kurzke, and B. A. Miller, “Device and Service Discovery in Home Networks with OSGi,” *IEEE Communications Magazine*, Vol. 40, Issue 8, pp.86-92, August 2002
5. K. Kang and J. Lee, “Implementation of Management Agents for an OSGi-based Residential Gateway,” *The 6th International Conference on Advanced Communication Technology*, Vol. 2, pp.1103-1107, 2004
6. F. Yang, “Design and Implement of the Home Networking Service Agent Federation Using Open Service Gateway,” *International Conference on Integration of Knowledge Intensive Multi-Agent Systems*, pp.628-633, Sept._Oct. 2003
7. H. Zhang, F. Wang, and Y. Ai, “An OSGi and Agent Based Control System Architecture for Smart Home,” *Proceedings of IEEE Networking, Sensing, and Control*, pp.13-18, March 2005
8. Knopflerfish. http://www.knopflerfish.org
9. K. Chen and L. Gong, Programming Open Service Gateways with Java Embedded Server^{TM} Technology, Addison Wesley, 2001
10. L. Gong, “A Software Architecture for Open Service Gateways,” *IEEE Internet Computing*, Vol. 5, Issue 1, pp.64-70, Jan.-Feb. 2001
11. R. S. Hall and H. Cervantes, “Challenges in Building Service-Oriented Applications for OSGi,” *IEEE Communications Magazine*, Vol. 42, Issue 5, pp.144-149, May 2004
12. R. S. Hall and H. Cervantes, “An OSGi Implementation and Experience Report,” *First IEEE Consumer Communications and Networking Conference*, pp.394-399, Jan. 2004

A Ubiquitous Workflow Service Framework

Joohyun Han, Yongyun Cho, Eunhoe Kim, and Jaeyoung Choi

School of Computing, Soongsil University,
1-1 Sangdo-dong, Dongjak-gu, Seoul 156–743, Korea
{jhhan, yycho, ehkim}@ss.ssu.ac.kr, choi@ssu.ac.kr

Abstract. In ubiquitous environments, all services head for context-awareness to provide appropriate services for a user's situation. However, it is hard to implement all kinds of things related to context managements. In this paper we propose a ubiquitous workflow service framework, named uFlow, based on a structural context model and uWDL, which is a ubiquitous workflow description language. Service developers can easily describe context-aware services using the uFlow framework so long as they only select available services based on Web Services and describe context information as a transition condition of workflow services. In order to verify the effectiveness of the uFlow framework, we designed and implemented a service scenario described with uWDL, and demonstrated that the scenario provides users with appropriate services according to a user's situation in ubiquitous computing environments.

1 Introduction

WfMC (Workflow Management Coalition) states that a workflow expresses flows of subtasks until a process is completed using standardized methods [1]. Between the subtasks in a workflow, there exist various relationships such as dependency, ordering, and concurrency. Workflows describe flows of subtasks using a workflow language. A workflow management system manages and controls flows of subtasks using state-transition constraints specified in the workflow language. Modeling a workflow can help software designers better understand how to support users when they design applications.

An application or a service that uses context information or performs context-appropriate operations is called a context-aware application or a context-aware service [2, 3]. In order to provide a context-aware service in ubiquitous environments, an appropriate service is selected and executed based on context information. Service developers should describe and handle context information to build context-aware services. However, it is so difficult to implement all sorts of things related to context managements such as context wrapper, context query system, ontology server, and so on.

In this paper, we propose a ubiquitous workflow service framework, named uFlow, based on a structural context model and uWDL. uWDL is a ubiquitous workflow description language and it can specify the context information on the transition conditions of workflow services to provide users with adaptive

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 30–39, 2006.

services for a user's current situation, and the structural context model is used to express context information in uWDL. Service developers can easily design context-aware services using the uFlow framework so long as they only select available services based on Web Services and describe context information as a transition condition of workflow services.

2 Related Work

Gaia [4] supports a service environment in which ubiquitous applications can communicate context information to each other. It depends on a specific protocol which is not widely used because it is based on CORBA middleware. LuaOrb, that is Gaia's script language, can instantiate applications and interact with execution nodes to create components and easily glue them together, but it can't express dependency or parallelism among the services because it describes only a sequential flow of specific services. The uFlow framework is a workflow system based on Web Services which are platform- and language-independent standard service interfaces, so it can express dependency and parallel execution among the services in heterogeneous ubiquitous computing environments.

BPEL4WS [5], WSFL [6], and XLANG [7] are Web Service-based workflow languages for business processes and distributed computing environments. They support service transition, and use XML-typed messages defined in other services using XPath. Context information is a complex data set that includes data types, values, and relations among the data types. XPath cannot sufficiently describe diverse context information because it can use only condition and relation operators to decide transition conditions. uWDL uses a context triplet - subject, verb, and object - in order to express high-level context information as transition conditions, which can not be supported by existing workflow languages.

3 The Key Components for uFlow

A context in ubiquitous computing environments indicates any information that is used to characterize the situation of an entity [3]. In ubiquitous environments, all services head for context-aware services to provide appropriate services for a user's situation. In order to provide context-aware workflow services in ubiquitous environments, an appropriate service is selected and executed based on context information. Therefore we designed two components, which are a structural context model and a ubiquitous workflow description language to use the context information as the constraints of the state transition in ubiquitous workflow services. The two components are designed based on knowledge structure to express the context information in a simple and flexible way.

3.1 Structural Context Model

A structural context model expresses ubiquitous context information from a viewpoint of knowledge structure. Because it has an information structure to express complex context information, it is possible to describe contexts to specify the context information on a transition condition of services in uFlow scenario documents. Figure 1 shows a class diagram of the structural context model. Context information can be any information that describes a situation of an entity. An entity is a person, place, physical, or logical thing which is considered in ubiquitous computing environment. We designed the structural context model which is an ontology-based context model for uFlow using OWL (Web Ontology Language), and it describes context information as entities having context types which have its values. The ontology language OWL builds on RDF (Resource Description Framework) [8], and a RDF statement is always a triple of resource, property and value, in that order. Because our model is based on ontology using OWL, our core concepts which are subject, verb, and object can be mapped into resource, property, and value on RDF, respectively.

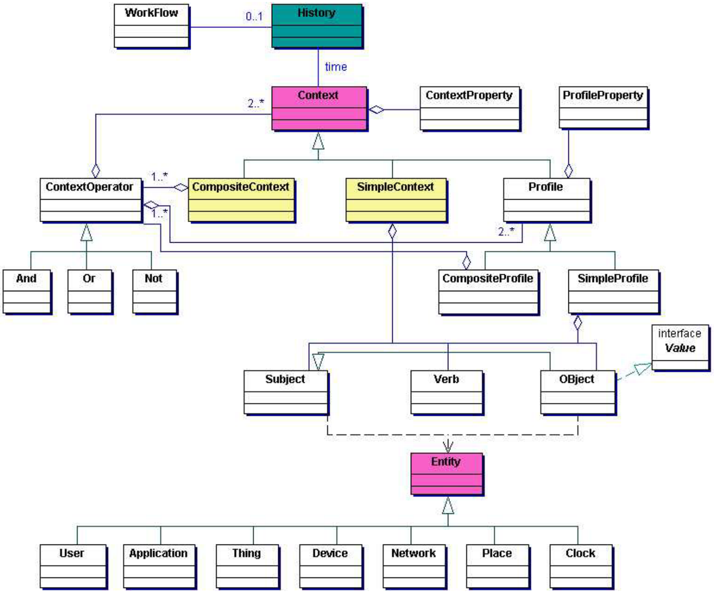


Fig. 1. Structural context model

3.2 Ubiquitous Workflow Description Language

A workflow management system manages and controls flows of subtasks using state-transition constraints specified in a workflow language. Although current workflow languages such as BPEL4WS, WSFL, and XLANG can specify the flows among services based on Web services, these workflow languages do not support the ability that controls the state-transition constraints using context, profile, or event information in ubiquitous computing environments. uWDL (Ubiquitous Workflow Description Language)[9] is a Web Services-based workflow language that describes service flows, and provides the functionalities to select an appropriate service based on high-level contexts, profiles, and events, which are obtained from various sources and structured by Ontology [10]. To provide these functionalities, uWDL specifies context and/or profile information as a triplet of {subject, verb, object} based on the structural context model for rule-based reasoning which can effectively represent the situation in a simple and flexible way. Figure 2 shows the schema structure of uWDL.

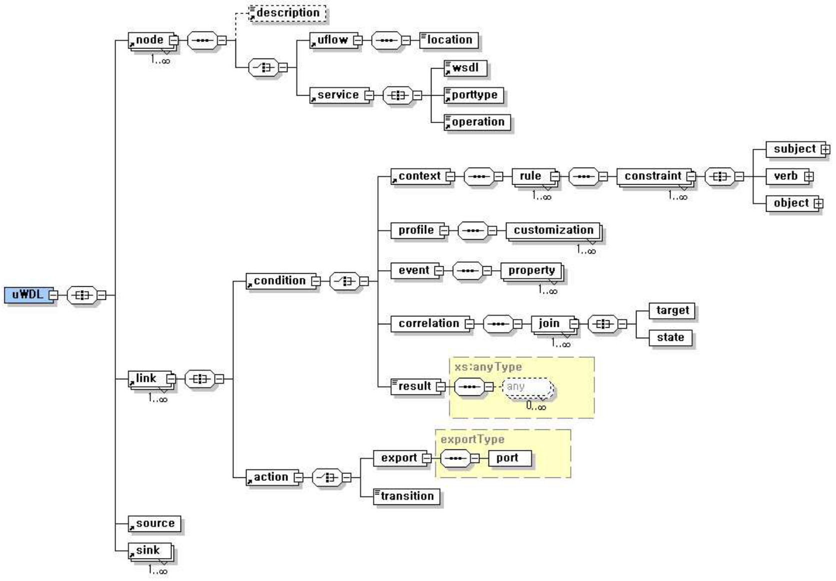


Fig. 2. uWDL schema

4 Ubiquitous Workflow Service Framework

Service developers have to describe and manipulate context information for context-awareness of services. However, they have a great difficulty in implementing all kinds of things related to context managements such as context wrapper, context query system, ontology server, and so on. We propose a ubiquitous workflow service framework named uFlow based on the key components in Section 3. Service developers can easily design context-aware services using the uFlow framework so long as they only select available services based on Web Services and describe context information as a transition condition of workflow services. The uFlow framework consists of uFlow scenario editor, uFlow engine, and uFlow context processor as shown in Figure 3.

A service developer can use the uFlow scenario editor to create a scenario document written with uWDL. The service developer can query context information to the ontology server through context browser to obtain standard vocabularies of context information for a specific domain and specifies the context information as a transition condition of workflow services. The scenario document created

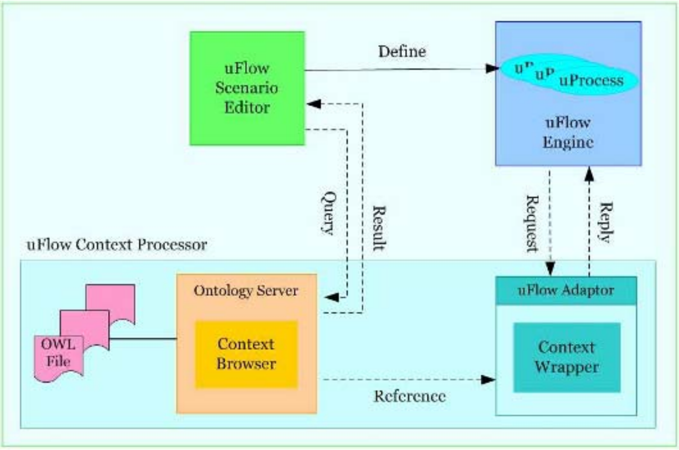


Fig. 3. uFlow framework

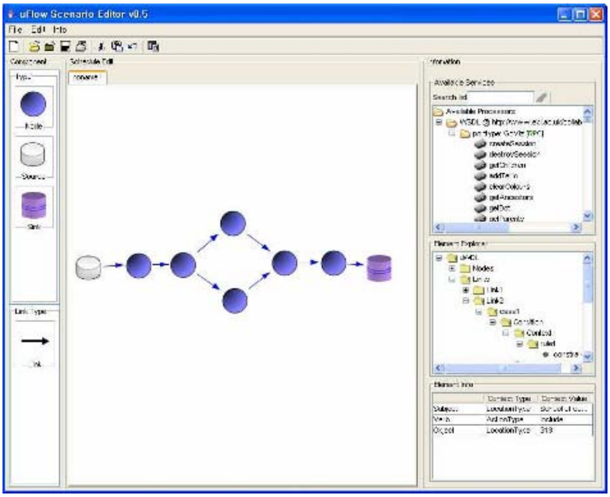
by uFlow scenario editor is delivered into uFlow engine in order to parse and manipulate the context information according to the service flows. The uFlow engine collects context information through a context wrapper in uFlow context processor and compares these context information with the context information described in the scenario document. If the matching result is true, an appropriate service is executed by the uFlow engine. The detailed explanations are as follows.

4.1 uFlow Scenario Editor

uFlow scenario editor is a tool for developers to easily design scenario documents without detailed understanding of uWDL schema. Developers can select currently available services based on Web Services and describe context information as a transition condition of the services. The uFlow scenario editor provides drag and drop capabilities to <node> and <link> elements in uWDL and consists of available services, element explorer, and context information obtained from current sensing environments. A scenario document is created by uflow scenario editor, and translated and executed by uFlow engine. Figure 4 shows uFlow scenario editor.

4.2 uFlow Engine

A uWDL document designed for a specific scenario should be translated and executed to provide adaptive services for a user's situation. For this purpose, we need a process to manipulate contexts aggregated from a sensor network. Figure 5 shows uFlow engine for handling context information expressed in uWDL. uWDL parser parses a uWDL scenario document and produces a DIAST (Document Instance Abstract Syntax Tree) [11] as a result. A DIAST represents the syntax of a scenario document, and is used to compare contexts expressed



	Context Type	Context Value
Subject	LocationType	[illegible]
Verb	ActionType	include
Object	LocationType	313

Fig. 4. uFlow scenario editor

in a scenario with entities aggregated from a sensor network to verify their coincidence. A context is described by one or more constraint elements, and each constraint is represented by a context triplet of {subject, verb, object} in a sequence. In Figure 5, a partial subtree in dotted lines indicates a subtree that makes up context constraints in the scenario.

A context mapper extracts types and values from objectified entities aggregated from a sensor network, and composes a subtree which consists of subject, verb, and object information. It then compares the type and the value of an entity with those of the constraint element in the DIAST subtree, respectively. If the type of the entity matches with its counterpart in the constraint element, the context mapper regards it as a correct subelement of the constraint element. If each entity has the same type, it may be ambiguous to decide the context's constraint according to its entity type only. The problem can be resolved by comparing the value of the objectified entity with that of the constraint element in the DIAST subtree.

4.3 uFlow Context Processor

uFlow context processor takes a responsibility of providing context information with uFlow scenario editor and uFlow engine. uFlow context processor consists of context browser, context wrapper and ontology server. Usually a Context infrastructure treats low-level context information that is raw contextual information which comes from sensors such as temperature, noise level, and location. However, uFlow needs not only low-level context information but also high-level context information that is combined by two or more low-level context information. Ontology server has functions which reason high-level context information from some low-level context information, and which reason explicit

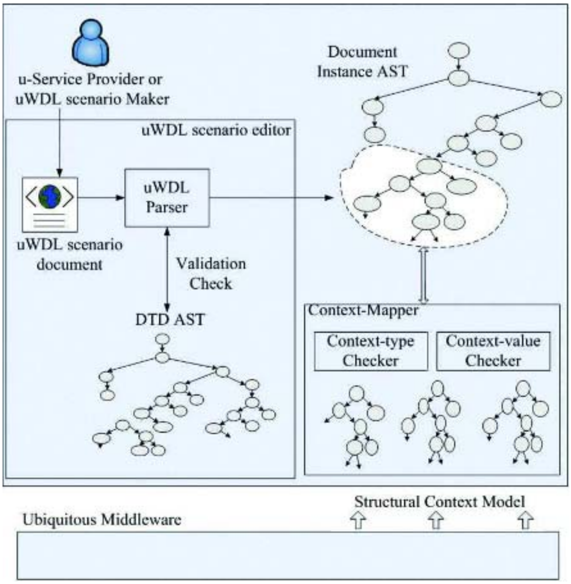


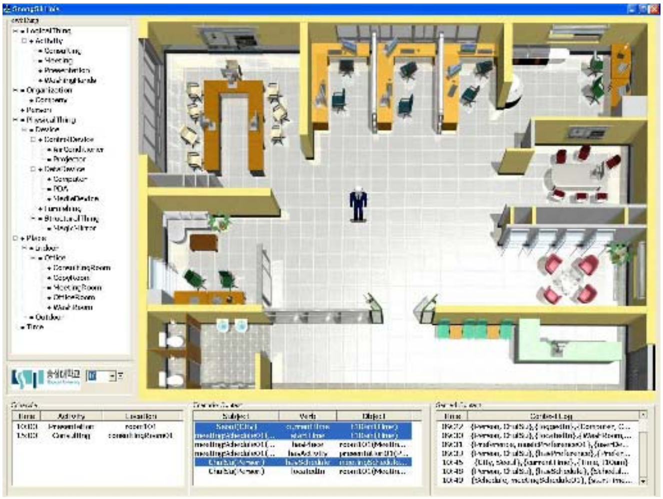
Fig. 5. uFlow engine

context information from implicit context information through our Ontology-based context model and ontology reasoner.

Context wrapper transforms context information obtained from a sensor network into a form of structural context model adequate to uFlow Engine. Context information structured using the structural context model consists of a enity(subject), a type of the entity(verb), and a type of the value(object). uFlow scenario editor is a tool which defines a sequence of services using context information. uFlow scenario editor requests context browser to browse available context information provided by ontology server, and context browser delivers them to uFlow scenario editor.

5 Experiments

In this section, we show a process to decide a state transition condition according to context information. For testing, we simulated a ubiquitous office in Figure 6 using uFlow framework. The purpose is "implementing a service which prepares an office work automatically according to a user's situation." Context information is simulated in a virtual office environment based on GUI according to a variation on the schedule, time, and/or user's location and preference. These context information is structured using the structural context model in Section 3 and transmitted to uFlow engine in order to identify current situation. uFlow engine executes related services which exist in a form of Web Services according to the user's situation. The scenario context tab in Figure 6 shows the progress of uFlow engine how to select a service according to dynamically incoming context information.

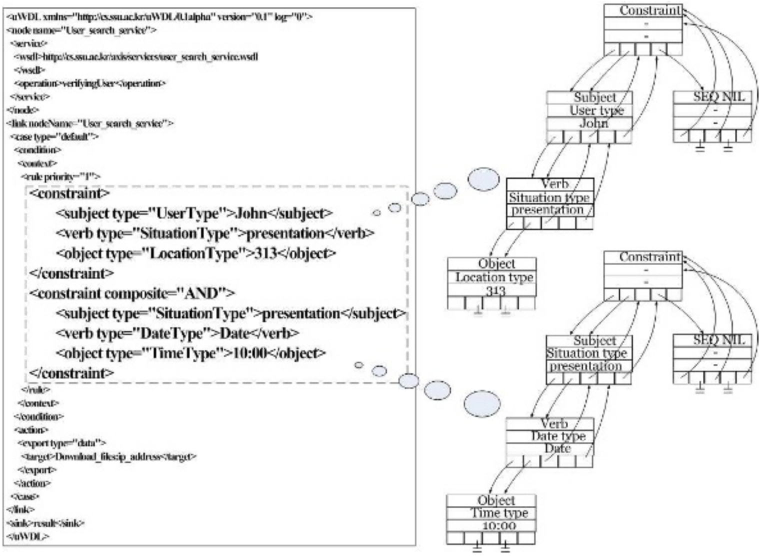


Time	Activity	Location
10:00	Presentation	room101
15:00	Consulting	consultingRoom01

Subject	Verb	Object
Seoul(City)	currentTime	[illegible]
meetingSchedule01(...	startTime	[illegible]
meetingSchedule01(...	hasPlace	room101(Meetin...
meetingSchedule01(...	hasActivity	presentation01(P...
ChulSu(Person)	hasSchedule	meetingSchedule...
ChulSu(Person)	locatedIn	room101(Meetin...

Time	Context Log
[illegible]	{Person, ChulSu}, {loggedIn}, {Computer, C...
[illegible]	{Person, ChulSu}, {locatedIn}, {WashRoom,...
[illegible]	{Preference, [illegible]}, {user[illegible]...
[illegible]	{Person, ChulSu}, {hasPreference}, {Prefer...
10:45	{City, Seoul}, {currentTime}, {Time, 10am}
10:49	{Person, ChulSu}, {hasSchedule}, {Schedul...
10:49	{Schedule, meetingSchedule01}, {start time...

Fig. 6. The Simulation of a Ubiquitous Office



```
<uWDL xmlns="http://cs.ssu.ac.kr/uWDL/0.1alpha" version="0.1" log="0">
<node name="User_search_service">
 <service>
  <wsdl>http://cs.ssu.ac.kr/axis/serviers/user_search_service.wsdl
  </wsdl>
  <operation>verifyingUser</operation>
 </service>
</node>
<link nodeName="User_search_service">
 <case type="default">
  <condition>
   <context>
    <rule priority="1">
     <constraint>
          <subject type="UserType">John</subject>
          <verb type="SituationType">presentation</verb>
          <object type="LocationType">313</object>
     </constraint>
     <constraint composite="AND">
          <subject type="SituationType">presentation</subject>
          <verb type="DateType">Date</verb>
          <object type="TimeType">10:00</object>
     </constraint>
    </rule>
   </context>
  </condition>
  <action>
   <export type="data">
    <target>Download_files:ip_address</target>
   </export>
  </action>
 </case>
</link>
<sink>result</sink>
</uWDL>
```

Fig. 7. A scenario document and a DIAST's subtree produced by uWDL parser

Figure 7 shows a scenario document designed using the uFlow scenario editor. If uFlow engine receives context data objectified as (SituationType, presentation), (UserType, Michael), (UserType, John), and (LocationType, 313), it compares the contexts' types and values with the subtree's elements. In this case, the context (UserType, Michael) is not suitable for anywhere in the subtree's elements, so uFlow engine removes the context. For the experiment, we named a

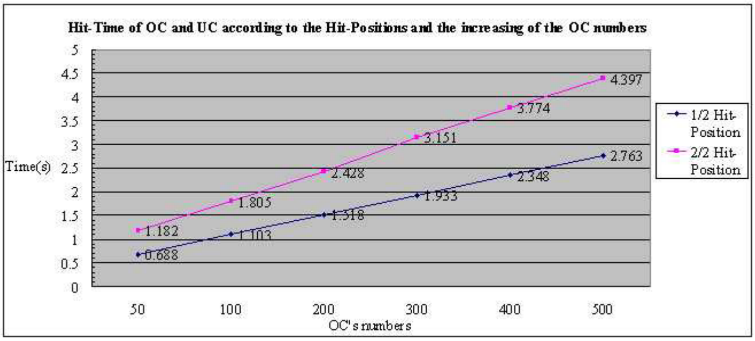


Fig. 8. A hit-time for hit-position and the number of OCs

context of a scenario document as UC(uWDL Context) and a context obtained from a sensor network in ubiquitous computing environments as OC(Objectified Context). uFlow engine decides a service transition through a comparison between UCs and OCs. A context described in the scenario document consists of a limited number of UCs. On the other hand, contexts obtained from a sensor network can be produced as innumerable OCs according to a user's situation. Therefore, uFlow engine should select quickly and correctly an OC coinciding with a UC from such innumerable OCs.

In Figure 8, we generated a lot of OCs incrementally, and measured how fast the suggested uFlow engine found the OC of the produced OCs that coincided with the UCs of the scenario document shown in Figure 7. To get the hit-time, we placed the OCs coinciding with the UCs in the middle and the end of the OCs that we produced randomly. We used a Pentium 4 2.0 GHz computer with 512MB memory based on Windows XP OS for the experiment. We increased the OC's amounts by 50, 100, 200, 300, 400, and 500 incrementally.

In Figure 8, 1/2 hit-position means the position of the OC coinciding with the UC is the middle of the randomly produced OCs, and 2/2 hit-position means the position of the OC is the end of the randomly produced OCs. As shown in the result, the hit-time is not increased greatly regardless of the OCs's considerable increase. This result shows that the suggested uFlow engine can sufficiently support context-aware services.

6 Conclusion

In this paper, we proposed a uFlow framework for ubiquitous computing environments. The uFlow framework was designed based on uWDL which can easily describe service flows and the structural context model to express context information in uWDL. uWDL can specify the context information on transition constraints of a service workflow in ubiquitous computing environments, and is designed based on Web services. The uFlow framework consists of uFlow scenario editor, uFlow engine, and uFlow context processor. It is able to integrate, manage, and execute various heterogeneous services in ubiquitous environments.

Therefore, uFlow framework provides users with appropriate services according to the user's context information. We developed a scenario described with uWDL, and we demonstrated that the uFlow framework can provide users with autonomic services in ubiquitous computing environments. In the near future, we will expand uWDL schema to express more detailed situations by assigning semantic information to Web services.

Acknowledgements

This work was supported by the Ubiquitous Autonomic Computing and Network Project, funded by the Korean Ministry of Information and Communication (MIC).

References

1. D. Hollingsworth: The Workflow Reference Model. Technical Report. TC00–1003. Workflow Management Coalition (1994)
2. Guanling Chen, David Kotz: A Survey of Context-Aware Mobile Computing Research, Technical Report, TR200381, Dartmouth College (2000)
3. Anind k. Dey: Understanding and Using Context, Personal and Ubiquitous Computing. Vol 5. Issue 1. (2001)
4. Manuel, Roman, Christopher, K.: Gaia: A Middleware Infrastructure to Enable Active Spaces. IEEE Pervasive Computing (2002) 74–83
5. Tony, Andrews, Francisco, Curbera, et al.: Business Process Execution Language for Web Services. BEA Systems. Microsoft Corp. IBM Corp., Version 1.1 (2003)
6. Frank, Leymann: Web Services Flow Language (WSFL 1.0), IBM (2001)
7. Satish, Thatte: XLANG Web Services for Business Process Design, Microsoft Corp. (2001)
8. W3C: RDF/XML Syntax Specification, W3C Recommendation (2004)
9. Joohyun Han, Yongyun Cho, Jaeyoung Choi: Context-Aware Workflow Language based on Web Services for Ubiquitous Computing, ICCSA 2005, LNCS 3481, pp.1008–1017, (2005)
10. Deborah, L., McGuinness, Frank, van, Harmelen, (eds.): OWL Web Ontology Language Overview, W3C Recommendation (2004)
11. Aho, A., V., Sethi, R., Ullman, J., D.: Compilers: Principles, Techniques and Tools. Addison–Wesley (1986)

Self Organizing Sensor Networks Using Intelligent Clustering

Kwangcheol Shin, Ajith Abraham, and Sang Yong Han*

School of Computer Science and Engineering,
Chung-Ang University, 221, Heukseok-dong,
Dongjak-gu, Seoul 156-756, Korea
kcshin@archi.cse.cau.ac.kr, ajith.abraham@ieee.org,
hansy@cau.ac.kr

Abstract. Minimization of the number of cluster heads in a wireless sensor network is a very important problem to reduce channel contention and to improve the efficiency of the algorithm when executed at the level of cluster-heads. This paper proposes a Self Organizing Sensor (SOS) network based on an intelligent clustering algorithm which does not require many user defined parameters and random selection to form clusters like in Algorithm for Cluster Establishment (ACE) [2]. The proposed SOS algorithm is compared with ACE and the empirical results clearly illustrate that the SOS algorithm can reduce the number of cluster heads.

1 Introduction and Related Research

Research in wireless sensor networks has been growing rapidly along with the development of low-cost micro devices and wireless communication technologies [1]. Some of the research related to scientific, medical, military and commercial usage has gone to the background [4].

Sensor networks are composed of hundreds to myriads of sensor nodes, which appear to be sprinkled randomly by a car or airplane. Each node has strict limitation in the usage of electric power, computation and memory resources. They typically utilize intermittent wireless communication. Therefore, sensor networks should be well-formed to achieve its purposes. Clustering is a fundamental mechanism to design scalable sensor network protocols. The purpose of clustering is to divide the network by some disjoint clusters. Through clustering, we can reduce routing table sizes, redundancy of exchanged messages, energy consumption and extend a network's lifetime. By introducing the conventional clustering approach to the sensor networks provides a unique challenge due to the fact that cluster-heads, which are communication centers by default, tend to be heavily utilized and thus drained of their battery power rapidly. Algorithm for Cluster Establishment (ACE) [2] clusters the sensor network within a constant number of iterations using the node degree as the main

* Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 40 – 49, 2006.

parameter. Some of the weaknesses of ACE are: First, ACE randomly selects candidate node in each iteration which creates different results each time on the same sensor network. Second, spawning threshold function is used in ACE to control the formation of new cluster by using two manually adjusted parameters. ACE performance relies on these parameters which are usually manually adjusted according to the size and shape of a sensor network.

In the literature, besides ACE, there are some related works on forming and managing clusters for sensor networks. For example, LEACH [5] rotates the role of a cluster head randomly and periodically over all the nodes to prevent early dying of cluster heads. Guru et al. [6] consider energy minimization of the network as a cost function to form clusters. Mhatre and Rosenberg [7] take into account not only the battery of the nodes but also the manufacturing cost of hardware.

In this paper, we propose a new clustering algorithm that does not require manually adjusted parameters which could also provide identical results in each test on the same sensor network to overcome the weakness of ACE. Rest of the paper is organized as follows. In Section 2, we present the clustering problem followed by Section 3 wherein the new algorithm is illustrated. Experiment results are presented in Section 4 and some conclusions are also provided towards the end

2 The Clustering Problem

Clustering problem can be defined as following. Assume that nodes are randomly dispersed in a field. At the end of clustering process, each node belongs to one cluster exactly and be able to communicate with the cluster head directly via a single hop [3]. Each cluster consists of a single cluster head and a bunch of followers as illustrated in Figure 1. The purpose of the clustering algorithm is to form the smallest number of clusters that makes all nodes of network to belong to one cluster. Minimizing the number of cluster heads would not only provide an efficient cover of the whole network but also minimizes the cluster overlaps. This reduces the amount of channel contention between clusters, and also improves the efficiency of algorithms that executes at the level of the cluster-heads.

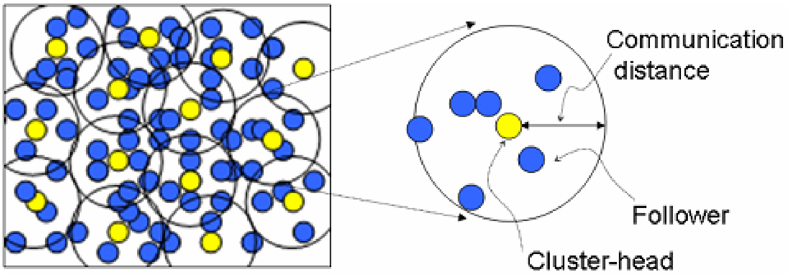


Fig. 1. Clustering in a sensor network

3 Self Organizing Sensor (SOS) Networks by Minimization of Cluster Heads Using Intelligent Clustering

3.1 Global Level of Clustering Algorithm

This Section presents the proposed clustering algorithm in a global scale, and the following section describes the algorithm at a node level. The following steps illustrate an overview of the suggested algorithm.

1. *Find the node (No), which has the maximum number of followers, and make a cluster with it.*
2. *Include clustered nodes into a clustered node set G.*
3. *Selects the next head node (Nf), which can communicate with a node in G and has the maximum number of followers, and make a cluster with it.*
4. *If there exist an unclustered node or nodes then go to step 2*
5. *Else terminate the algorithm.*

At first, it makes a cluster with the center node which has the maximum number of followers. We assume that there is a coordinator which controls globally in the entire network (for easy understanding). So it does not matter to locate the center node during step 1. In step 2, it includes the selected cluster head node and its followers to the clustered node set *G*. And in step 3, it selects the node, which can communicate with a node in *G* and has the maximum number of followers, and makes a cluster with it as a cluster head and include it and its follower to set *G*. Figure 2 illustrates step 3. A node *'a'* is *'No'* node and node *'b'* is the node which can communicate with the next head node (that is, node *'c'*), which has the maximum number of followers. Then it elects node *'c'* as a next cluster head node and makes a cluster with it. The process is then repeated until all the nodes are clustered.

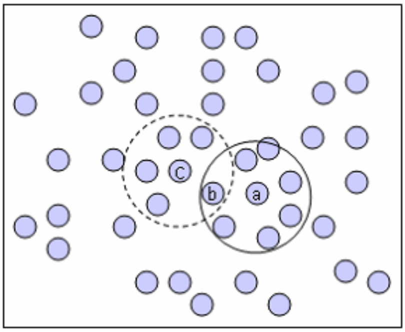


Fig. 2. Clustering example

3.2 Node Level of Clustering Algorithm

Node level algorithm is mainly divided into two parts, first part for finding out a node which has the most number of followers and makes it as the first cluster head, and the second one for the actual clustering process.

Table 1. Message and methods

Message Structure: (command, data, node_id)
Methods description :
broadcast(message) : send a message to everyone which it can communicate with
send(message, destination) : send a message to a destination

To implement the algorithm, we introduce '*message*' which has three parts, (*command*, *data*, *node_id*) and two methods which are used frequently, broadcast (*message*), which sends a message to everyone, to which it can communicate with and send (*message*, *destination*) which sends a message to a destination. The concept of *message* and *methods* is illustrated in Table 1.

We also define two concepts:

Super-node: The node which is selected as a head of first cluster, to decide which node will be the new cluster head (for example, node *'a'* in Figure 2).
Linker node: The node which communicate between two cluster heads. This node is included in two clusters which it connects (for example, node *'b'* in Figure 2).

Table 2. Algorithm for finding the super-node

```
myState := Super_Head
n := number of my neighbors
c := myID
while (myState is Super_Head and c is not 0)
        c := c -1
        if (notEmpty(msgQueue)
                message := find_best_one(msgQueue)
                if(message.data >= n)
                        myState := Unclustered
                        broadcast(message)
if (myState is Super_Head) broadcast(( ,n, ))
d := n
t := sufficient time +  myID
While (t is not 0)
        t := t-1
        message := wait_for_a_message()
        if (message.data > n)
                myState := Unclustered
                if(d < message.data)
                        d := message.data
                        broadcast(message)
if (myState is Super_Head) broadcast(("recruit", ,myID))
Purge(msgQueue)
```

3.2.1 Discovery of Nodes Which Has the Most Followers

To find the node which has the maximum number of followers, we suggest a method as illustrated in Table 2. In the first stage, state of every node is considered as a super-head. Each node counts the number of its neighbors and it sets variable *c* as its unique identification number (ID) to execute the algorithm one by one without collisions. This unique ID for the individual sensors is decided when the sensors are spread.

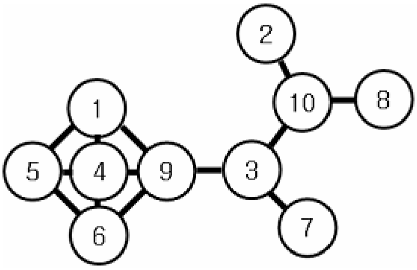


Fig. 3. Illustration for finding the super-node

For the sensor network illustrated in Figure 3 (the number in each circle is a unique ID for each sensor), we present how the proposed algorithm could set up node 4 as a super-node. It is important to remember that each node performs its own algorithm operation independently to setup the super-nodes. At first, node 1 sends message to its neighbors and nodes 5, 4 and 9 will receive the message which node 1 sent. Message queue of nodes 5, 4 and 9 are shown in Figure 4 and node 1 will get into the state of waiting for a message. After that, node 2 broadcasts its number of neighbors to its neighbor node 10, and node 3 to nodes 7, 9 and 10. Node 2 and 3 will also get into the state of waiting for a message as shown in Figure 5.

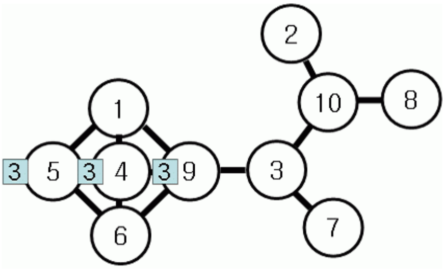


Fig. 4. Message queue of nodes 5, 4 and 9 after node 1 broadcasts

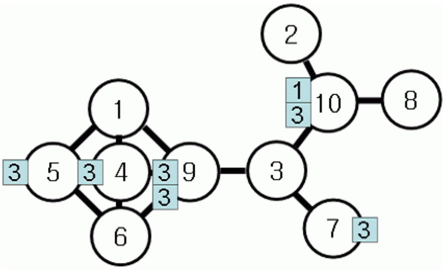


Fig. 5. State after nodes 2 and 3 broadcast a message

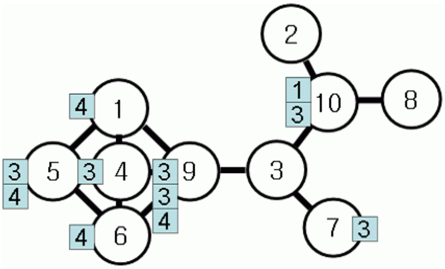


Fig. 6. After node 4 broadcasts a message

By turn, node 4 performs its operation and its message queue is not empty. So, node 4 finds the message, which has the biggest data value, in its message queue and compares it with its number of neighbors. In this case, node 4's number of neighbors is 4 and the biggest one in message queue is 3, so node 4 broadcasts its number of neighbors as shown in Figure 6. Node 1 will now receive the message, which node 4 sent, and it changes its status as unclustered since arrived *'message.data'* is bigger than its number of neighbors and broadcast arrived *'message.data'* again. The procedure is illustrated in Figure 7.

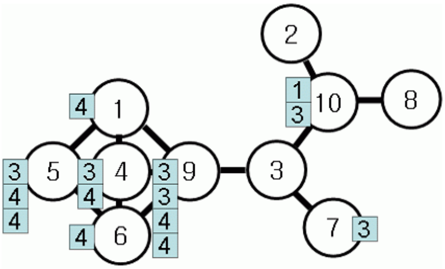


Fig. 7. After node 1 broadcasts a message which it received

Node 5 executes its algorithm and the number of its neighbors is 3 and the biggest one in message queue is 4, so it changes its status as unclustered and broadcasts *'message.data'*, which is 4. Node 6 executes its algorithm and its number of followers is smaller than the biggest one in queue, and it changes its status as unclustered and broadcasts the biggest *'message.data'*. After doing all of procedures, node 4 will remain as super-node and all of rest will be unclustered status. And finally, node 4 broadcasts a recruit message to its neighbors to make a cluster with node 4 as cluster head.

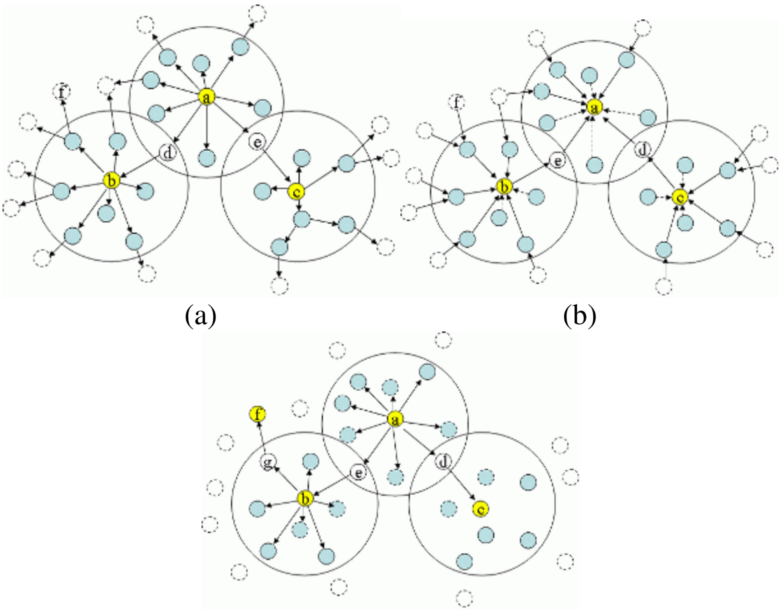
3.2.2 Self Organizing Sensor (SOS)Clustering Algorithm

Table 3 illustrates the pseudo code of the SOS clustering algorithm and it consists of 5 parts.

Table 3. SOS clustering algorithm

```
myHead := NONE   // my cluster head
nextHead := NONE   // for linker node, which has two head

// for unclustered node
while (myState is Unclustered)
        message := wait_for_a_message()
        if (message.command is "survey")
                uf := calculate_number_of_followers(myID)
                send(("report",uf,myID),message.node_id)
        if (message.command is "recruit")
                myHead := message.node_id
                myState := Clustered
        if (message.command is "notify" and message.node_id is myID)
                myState := Cluster_Head
                broadcast(("recruit", ,myID))

// for clustered node
while (myState is Clustered)
        message := wait_for_a_message()
        followers := NONE        // array for follower nodes
        if (message.command is "survey")
                followers := update_my_followers(myID)
                if(followers is not NONE)
                        send(("survey", ,myID),followers)
                        msgQueue := wait_for_followers_reports()
                        nodeBest := find_best_node(msgQueue)
                        message := (message.command,message.data,myID)
                        send(nodeBest,myHeader)
                        purge(msgQueue)
                 else
                        send(("report",NONE,NONE),myHead)
                        terminate()
        if (message.command is "notify" and message.node_id is myID)
                myState := Linker
                nextHead := nodeBest.node_id
                send(message,nodeBest.node_id)

// for super-head
while (myState is Super_Head)
        broadcast(("survey", , ))
        msgQueue := wait_for_followers_reports()
        networkBest := fine_best_node(msgQueue)
        if(networkBest.node_id is NONE)      terminate()
        else broadcast_to_follwers(("notify",,networkBest.node_id))
        purge(msg_queue)

// for cluster head
while (myState is a Cluster_Head)
        message := wait_for_a_message()
        if (message.command is "survey")
                broadcast_to_followers("survey", ,myID)
                msgQueue : = wait_for_followers_reports()
                clusterBest := find_best_node(msgQueue)
                send(clusterBest,message.node_id)
                if (clusterBest.node_id is NONE) terminate()
                purge(msgQueue)
        if(message.command is "notify" and message.node_id is clusterBest.node_id)
                broadcast_to_followers(message)

// for linker
while (myState is a Linker)
        message := wait_for_a_message()
        if (message.command is "survey")
                message.node_id = myID
                send(message, nextHead)
        if (message.command is "notify") send(message, nextHead)
        if (message.command is "report")
                send(message, myHead)
                if(message.node_id is NONE) teminate()
```

The clustering process is illustrated in Figure 8. Every node, whose status is unclustered, waits for a message. The super-node (node *'a'* in Figure 8) broadcasts *'survey'* message to its followers. Every node, which receives *'survey'* message from its cluster head (include super-node), investigates that how many unclustered nodes exist within the area of its communication range. If there are no existing nodes that can communicate with, then it reports it to their head and terminates its algorithm. If some



(a) (b)

(c)

Fig. 8. Illustration of **(a)** *'survey'* process **(b)** *'report'* process and **(c)** *'notify'* process

nodes exist, it send *'survey'* message to every follower and waits for its *'report'* messages as shown in Figure 8. When every follower reports about it, the node selects follower's ID, which has the biggest number of neighbors, and save that follower's ID and sends a *'report'* back to its head recursively. This works in a recursive way and every *'report'* message arrives in super-node (Figure 8-b). If super-node get all report from every follower, then it selects a message contains the follower's id, which has the biggest number of neighbors, and broadcasts *'notify'* message with that follower's ID to its followers. Every clustered nodes, which receive *'notify'* message, compares *'notify.node_id'* with saved id and if it is same, then it changes its status as *'linker'* and set its next-head as saved node id, and sends a *'notify'* message to its next-head. If cluster-head received a *'notify'* message, then it compares *'notify.node_id'* with stored ID and if it is same then it broadcasts otherwise just drop it. If unclustered node received *'notify'* message then it changes its status as cluster-head and broadcasts a *'recruit'* message to its followers to make a cluster with it. If super-head get every *'report'* message with *'none'* then it terminates its algorithm (Figure 8-c).

Table 4. Test results for 2500 nodes

Case of 2500 nodes (500*500 rectangle space)			
Communication distance of a node	Number of generated clusters		Improvement ((ACE-SOS)/ACE)
	ACE (k_1=2.3,k_2=0.1)	SOS	
30	308	255	17.21%
50	126	114	9.52%
70	68	59	13.24%
100	38	35	7.89%
Average			11.97%

4 Experiment Results

The proposed SOS algorithm was implemented and compared with the ACE algorithm. We randomly scattered 2500 nodes in a 500*500rectangle space. Table 4 illustrates the performance results for 2,500 nodes. For comparison purposes, we set the communication range of each node as 30, 50, 70 and 100. In case of ACE, we manually adjusted k_1 and k_2 to achieve the best results. As shown in Table 4, the number of cluster heads could be reduced by about 11.97% (average) for 2,500 nodes when compared to the ACE approach. By using the SOS approach, we can efficiently reduce the routing table sizes, redundancy of exchanged messages, energy consumption and extends the network's lifetime.

5 Conclusions

In this paper, we presented a new clustering algorithm for minimizing the number of cluster heads. The proposed algorithm produces identical results every time for same network without using any network dependent parameters. Empirical results clearly show that the SOS algorithm could reduce the number of cluster heads by about 11.97% for 2,500 nodes when compared to the ACE approach.

Although our algorithm efficiently formulated the required clusters, there are several things to consider such as problems related to fast dying cluster heads and so on. We are also planning to incorporate more heuristic techniques to make the clustering process more efficient.

Acknowledgements

Work supported by the MIC (Ministry of Information and Communication), Korea, under the Chung-Ang University HNRC-ITRC (Home Network Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).

References

1. J.M. Kahn, R.H. Katz and K.S. Pister, Next Century Challenges : Mobile Networking for "Smart Dust", Proceedings of Mobicom, August 1999.
2. H. Chan, A. Perrig, ACE: An Emergent Algorithm for Highly Uniform Cluster Formation. In 2004 European Workshop on Sensor Networks. pp. 154-171.
3. Younis and S. Fahmy. Distributed Clustering in Ad-hoc Sensor Networks: A Hybrid, Energy-Efficient Approach. In Proceedings of IEEE INFOCOM, March 2004.
4. I.F. Akyildiz, W. Su, Y. Sankarsubramaniam and E. Cayirci, “Wireless Sensor Networks : a survey”, Computer Networks, Vol. 38, pp. 393-422, March 2002.
5. W. Heinzelman, A. Chandrakasan and H. Balakrishnan, “An Application Specific Protocol Architecture for Wireless Microsensor Networks,” IEEE Transactions on Wireless Communications, Vol. 1, No. 4, October 2002.
6. S. M. Guru, A. Hsu, S. Halgamuge, S. Fernando, “An Extended Growing Self-Organizing Map for Selection of Clusters in Sensor Networks”, International Journal of Distributed Sensor Networks, Volume 1, Number 2 / April-June 2005
7. V. Mhatre and C. Rosenberg, “Homogeneous vs. Heterogeneous Clustered Sensor Networks: A Comparative Study”, 2004 IEEE International Conference on Communications (ICC 2004), Paris France, June 2004.

Searching and Selecting Web Services Using Case Based Reasoning

Olivia Graciela Fragoso Diaz[1], René Santaolaya Salgado[1], Ismael Solís Moreno[1], and Guillermo Rodríguez Ortiz[2]

[1] Centro Nacional de Investigación y Desarrollo Tecnológico,
Interior Internado Palmira s/n Col. Palmira,
Cuernavaca, Morelos, México
{ofragoso, rene, isolis04c}@cenidet.edu.mx
[2] Instituto de Investigaciones Eléctricas,
Reforma 113, Edif. 27-1, Col. Palmira,
Cuernavaca, Morelos, México
gro@iie.org.mx

Abstract. Web services are currently one of the main technologies employed to create a systematic and extensible framework for application development. This is done by means of allowing the interaction among the applications of an organization. However, due to the large number of web services that may exist nowadays, locating one or several web services to fulfill the functional requirements of a user, an organization or a business entity, is a complex and time consuming activity for application developers. It also reduces their productivity. One possible solution for this problem is the implementation of a semantic component, structured as a library and populated with cases represented by web services in such a way that it may extend the functionality of the existing web services directories. The semantic component must provide a mechanism for classifying and selecting web services based on their functionality and supporting the search of WSDL description files of selected web services in a non sequential order within the directories. This paper describes a model for searching and selecting web services in UDDI directories supported by case based reasoning. Advantages and limitations of the model are also described.

1 Introduction

According to their functionality, web services are defined by [1] as "*software systems identified by a URL, whose public interfaces and bindings are defined and described using XML*".

Organizations may encapsulate their business processes and publish them as web services, subscribe to others and exchange information with other organizations. The model in which some companies participate with themselves in order to realize their business processes is getting to an end, because of the growing need to interact with other entities. Web services are also becoming the basis for electronic commerce. Organizations invoke other companies' services in order to complete their business transactions. In this context, how can organizations find or discover other companies

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 50 – 57, 2006.

to interact with in order to complete their business processes? if this is done by hand, no one can get the confidence of knowing all potential partners.

Same as any other internet resource, it is not possible to find a particular web service without the help of some tool that supports the searching activity. Nowadays this is done through interfaces provided by different operating nodes of web services registries, for which potential clients must know some precise data of the web service such as: the name of the service, categorization and name of the company. Other types of interfaces are similar to internet search engines, they use keywords and the result is a large list of web services. Because of this, in the web services discovery process, current searching methods are not enough, they lack precision in selecting required resources, affecting this way the relevance of the result, part of which is not of interest to the clients. Therefore, developers invest a considerable amount of time in understanding, discriminating and selecting those web services that are relevant for their application. According to this problem, it could be asked if technologies that facilitate the representation of domain knowledge, necessary to determine the similarity degree among the functionality of the web services useful for satisfying the clients requirements, such as the ones that employ case based reasoning "CBR" could improve searching precision and thus obtain smaller sets of relevant web services.

This paper proposes to employ CBR as a classification and selection mechanism to improve the precision in search results when looking for web services.

The model described in this paper uses structures and algorithms from case based reasoning as the underlying web semantic technology. This technology facilitates the representation of knowledge necessary to determine the similarity degree among the functionality of the web services stored in the library of cases and the requirements of the clients.

2 Case Based Reasoning CBR

In case based reasoning, the solution of past cases are remembered to solve a new problem. A case represents specific knowledge linked to a specific situation. It represents knowledge to an operational level; it makes explicit how to solve a task or how a piece of knowledge was applied in the solution of a problem [2]. In this context, web services are cases designed and implemented to solve specific problems. Because of their reusability property, they can be useful to solve new cases or particular situations.

The representation of organizational structures required in CBR and the translation of their implementation into the web context, requires the employment of techniques or tools that allow expressing the distinct association relationships among the elements of such structures. At the same time, they must provide the facilities to determine the similarity degree of the web services stored in a library of cases and the requirements of the new problem. Because of this, the use of the semantic web such as ontological languages for the representation of the structures is proposed, since establishing relationships among the elements allows enriching the structures in such a way that the determination of the similarity degree is less complex.

The CBR paradigm covers a range of different methods to organize and retrieve information, using indexed knowledge from past cases [3]. However, category schemes are those that have been applied mostly for retrieving and classifying components.

One of the first case based reasoning systems was developed by Kolodner [4] from the University of Yale. The system used a dynamic memory model that works basically as a query answer system. Bareiss and Porter [5, 6] proposed an alternative way for organizing the cases in a library using a network of categories. The psychological and philosophical base of this model is that natural concepts in the real world must be defined in such a way that they can be extended. In addition each category contains characteristics that define what cases may belong to it or what cases do not.

Case based reasoning has delivered satisfactory results in the acquisition and retrieval of information. Because of that, some CBR systems have been developed for critical domains. An example is the system developed by Plaza [3] for medical diagnosis and the system for selection and adjustment of valves for pipes on board [7] developed for General Dynamics in USA.

3 Semantic Web and Related Works

The objective of semantic web is to create a universal mean to interchange information by representing the meanings of the web resources in a format legible for machines. This pretends to widen the interoperability among the information systems and to reduce the intervention of human operators in the intelligent processes of information flows. It is expected that the semantic web helps to widen the capacity of the World Wide Web by using standards, markup languages and other tools applicable to their processes [9].

Several works on search and selection of web services have been developed; some of them employ web semantic techniques showing its applicability in the concrete description of the web resources.

Bernardi and Guninger [10] show a small set of test cases that motivate and illustrate the necessity for the creation of process models that can be interpreted by a computer and that enables the automatization of the searching and composition of web services. They propose that the process models be described as first order ontologies.

Mandell and MacIlraith [11] present an integrated technology for the customized and dynamic localization of web services, in addition to its interoperation through semantic conversion. They extend the BPWS4J with a mechanism denominated semantic discovery service “SDS” to provide semantic translation to match the user requirements.

Bilgin and Singh [12] developed a repository of web services that extends the UDDI (Universal Description Discovery and Integration) current search model. This repository combines an ontology of attributes with evaluation data. It is based on a language for manipulating the searching of web services based on DAML. The DAML language provides a wide variety of operations that are necessary for the maintenance of the ontology such as the publication of services, costs of services and services selection based on their functionality.

Benatallah [13] proposes a matching algorithm that takes as input the requirements to be met by the web services and an ontology of services based on logic descriptions. The outputs of the algorithm are the services that best comply with the requirements given as input to the algorithm.

Wang and Stroulia [14] developed a method for assigning a value of similarity on WSDL (Web Services Description Language) documents. The method may be used jointly with the API that accesses an UDDI in order to support an automatic process to locate web services, distinguishing among the services that can be potentially used and the services that are irrelevant to a given situation. The proposed method uses vector-space and WordNet to analyze the semantic of the identifiers of the WSDL documents in order to compare the structures of their operations, messages and types, and in this way to determine the similarity among two WSDL documents.

With the exception of Wang and Stroulia, most revised related works do not show experimental results. Therefore, a comparison among these works can not be objectively made.

4 Proposed Model for Searching and Selecting Web Services in UDDI Directories Supported by a Library of Cases

The model proposed in this paper is a case based reasoning model using category-exemplar organization, structured in such a way that allows service providers to describe their web services based on their functional characteristics. Functional characteristics are keywords relative to the domain and the function that the services perform. Also, the customer of the web services may search for the services based on functional characteristics, instead of on the company name, service or category name as UDDI actually does. The characteristics used in the library of cases are used either to index a new case or to locate a given web service. The model also requires a description of relevant characteristics to be used in the matching and selection process of the web services that cover most of the functional requirements given as input.

4.1 Components of the Model

The components of the proposed model are an UDDI, WSDL documents and a library of cases, which classifies the web services based on their functionalities. The library is supported on a module identified as *reasoner* in figure 1. The *reasoner* module contains two algorithms; one for locating the cases within the library and the other for matching the cases against the requirements of the user. The output is a set of cases classified in descendent order according to the level of similarity calculated by the algorithm. The model is shown in figure 1.

A library contains the knowledge of cases from a specific and well defined domain. The library has a structure known as discrimination network [2]. This network has a hierarchical organization whose objective is to group cases that are similar. The services of a group are the ones considered as the cases that best match the requirements input by the customer.

The algorithms of the *reasoner* module locate the cases directly on the appropriate places within the library, accessing only the cases that can be potentially used to solve the situation.

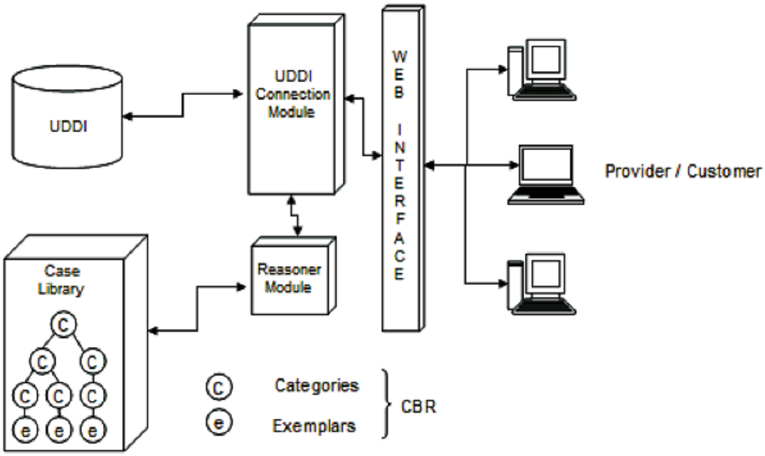


Fig. 1. Proposed model overview

The network is implemented using OWL (Ontology Web Language). The OWL allows representing the association relationships that exist among each of the components of the structure and the properties of each category and case. An example of a class definition using OWL is shown in the code below.

Example of the OWL code that implements a category in the structure of a library

```
<?xml version="1.0"?>
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-
ns#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns="http://www.owl-ontologies.com/unnamed.owl#"
  xml:base="http://www.owl-ontologies.com/unnamed.owl">
  <owl:Ontology rdf:about=""/>
  <owl:Class rdf:ID="Email">
    <Authentic
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >2</Authentic>
    <Validator
rdf:datatype="http://www.w3.org/2001/XMLSchema#int"
    >3</Validator>
    <rdfs:subClassOf rdf:resource="#Servicios"/>
  </owl:Class>
```

```
<owl:DatatypeProperty rdf:ID="Verify">
    <rdfs:domain rdf:resource="#Email"/>
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#int"/>
    <owl:equivalentProperty>
      <owl:DatatypeProperty rdf:ID="Check"/>
    </owl:equivalentProperty>
  </owl:DatatypeProperty>
```

4.2 How the Model Works

The proposed model respects the UDDI actual mechanism. While UDDI works as a universal directory, the model described here works as specific directory of a domain.

Figure 2 shows how the model works when a web service is registered. The first step is to provide the information required by the web interface (1) including the WSDL document and the functional characteristics of the web service. The connection module generates a unique identifier (2.1) and stores the information in the corresponding UDDI (2.2). When the service is registered in the UDDI, the unique identifier is sent to the reasoner module (3), which determines the category where the new case represented by the web service may be indexed based on the characteristics input in step (1) plus the unique identifier generated in step (2.1). When step 4 is completed, the web service registers in a UDDI and the library of cases is updated.

Figure 3 shows how the model works when a customer wants to search for a given service. First the customer must provide the requirements through the web

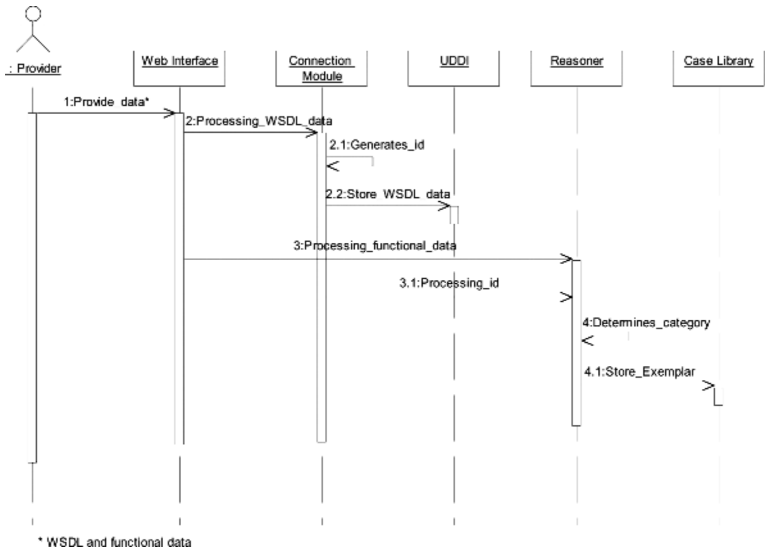


Fig. 2. Sequence to register a Web Service using the proposed CBR supported model

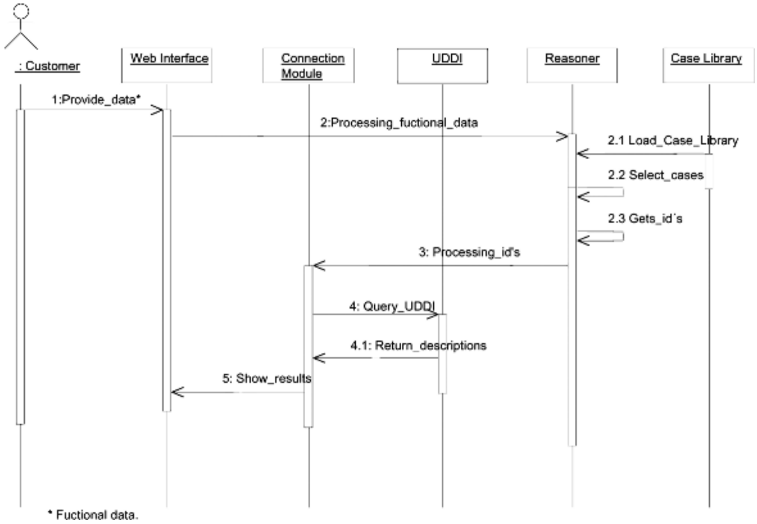


Fig. 3. Sequence to find a Web Service using the proposed CBR supported model

interface (1). The interface then sends the characteristics to the reasoner module (2). The reasoner module uses the characteristics of the required service and with the help of the searching and matching algorithms selects a set of cases relevant to cover the requirements specified by the customer. The reasoner also sends to the connection module the set of identifiers that reference the services as registered in UDDI (3). Then the connection module queries the UDDI (4) and shows the results in the web interface (5).

5 Advantages and Limitations

The model proposed in this paper seems to have some advantages against other related works. First, customers may obtain the service or set of services relevant to the problem that wants to be solved. Second, the requirements may be expressed by the customer in terms of the functionality of the web service needed, instead of for example the name of the service, the name of the organization that provides the service, or the categories specified in a UDDI. Third, this model provides a form to describe web services specifying the functionality they implement, and fourth, the model does not modify or eliminates the standards used nowadays for publishing and searching web services. They are only extended to offer a description of the functionality of the web services.

The model has also some limitations, the most important has to do with the fact that the library of cases works on well defined domains, the quality of the searching and indexing may be affected by the maturity of the domain employed to populate the library of cases. A solution to this problem is to encapsulate well defined application frameworks as web services. The lack of validation of the functional characteristics expressed by the service provider is also a limitation. This could be solved by means of matching algorithms such as the one employed by Wang and Stroulia [14], so that the veracity of the information given by the provider of the web services is tested.

Also by means of natural language comprehension algorithms to obtain the descriptions directly form the WSDL documents.

6 Conclusions

The UDDI model was developed to support the interaction among organizations. Although, it has a limited mechanism for publishing and searching web services, it is not necessarily a disadvantage since it may be extended with a search engine to return the web services that fulfills the user requirements. The extension to the UDDI model is the work described in this paper, and corresponds to a mechanism for searching and selecting web services based on category-exemplar type of CBR. CBR has shown to be useful for retrieving and classifying information of other domains, therefore it may be considered as the technique to extend the UDDI model.

Unfortunately, there is not a fair way to compare different approaches due to the lack of standard test cases. However, due to CBR advantages, it is expected that precision in searching and retrieving relevant web services be improved.

References

1. Dustdar, S., Shreiner, W.:A Survey on Web Services Composition. In: International Journal on Web and Grid Services. Vol. 1, No 1. (2005). pp 1-30.
2. Kolodner, J.: Case – Based Reasoning, Morgan Kaufmann Publisher Inc. (1993) 145-146
3. Aamondt, A., Plaza, E.: Case – Based Reasoning: Foundational Issues, Methodological Variations and System Approach. http://www.iiia.csic.es/People/enric/AICom.pdf
4. Kolodner, J.: Maintaining organization in a dynamic long-term memory. Cognitive Science. Vol. 7, 243 – 280. (1983)
5. Bareiss, R.: PROTOS; a unified approach to concept representation, classification and learning. Technical Report. (1998)
6. Porter, B., Bareiss, R., Holte, R.: Concep learning and heuristic classification in weak theory domains. Artificial Intelligence. Vol. 45. (1990) 229 -263
7. Brown, B., Lewis, L.: A case-based reasoning solution to the problem of redundant resolutions of nonconformance's in large scale manufacturing. In: R. Smith, C. Scott (eds.): Innovative Applications for Artificial Intelligence 3. MIT Press. (1991)
8. Wolfgang, S.: The State of Art of Searching the Web. German – American Frontiers of Engineering Symposium. (2004)
9. Wikipedia.: Web Semántica. http://es.wikipedia.org/wiki/Web_sem%C3%A1ntica
10. Berardi, D., Grüninger, M., Hull, R., McIlraith, S.: Towards a First – Order Ontology for Semantic Web Services. W3C WorkShop on Constraints and Capabilities for Web Services. (2004)
11. Mandell, D., McIlraith, S.: A Bottom – Up Approach to Automating Web Services Discovery, Customization, and Semantic Translation. Workshop on E-Services and the Semantic Web. (2003)
12. Soydan, A., Singh, M.: A DAML – Based Repository for QoS – Aware Semantic Web Service Selection. IEEE International Conference on Web Services. (2004)
13. Benatallah, B., Hacid, M., Leger, Alain., Rey, C., Toumani, F.: On Automating Web Service Discovery. VLDB Journal. Springer. (2004)
14. Wang, Y., Stroulia, E.: Semantic Structure Matching for Assessing Web – Service Similarity. First International Conference on Service Oriented Computing. (2003).

Fuzzy Logic Based Propagation Limiting Method for Message Routing in Wireless Sensor Networks[*]

Sang Hoon Chi and Tae Ho Cho

School of Information and Communication Engineering, Sungkyunkwan University,
300 Cheoncheon-dong, Jangan-gu, Suwon, Gyeonggi-do 440-746, Korea
{craken78, taecho}@ece.skku.ac.kr

Abstract. Recent advances in micro sensors and wireless communications have enabled the wireless sensor networks. Also, a number of routing protocols have been proposed for the sensor networks. Especially, the directed diffusion is a data-centric and application-aware routing algorithm in which all communication is processed by the attribute-value pairs of the named data. In the directed diffusion, an interest message is propagated through all the nodes within the network. However, the propagation to all the nodes is inefficient in terms of energy consumption. To solve this problem, we propose a new data propagation method in which the data transmission area is limited according to a threshold value for reducing the energy consumption in the network. The fuzzy rule based system is exploited to determine the threshold value by considering the energy and density of all the deployed nodes.

1 Introduction

Recent advances in MEMS (micro-electro-mechanical systems) and low power highly integrated digital electronics have enabled the development of low-cost sensor networks [1,2]. Wireless sensor networks consist of small nodes with sensing, computation, and wireless communications capabilities. Sensor nodes are usually scattered in a sensor field, which is an area where the sensor nodes are deployed. These sensor nodes have the ability to communicate either among each other or directly to an external base-station (BS) [3]. As a result, sensor networks have emerged as an important new tool for tracking contamination in hazardous environments, habitat monitoring in the nature preserves, enemy tracking in battlefield environments, etc [4]. In order to realize these various applications, it is necessary to solve the energy consumption problem and meet the requirements. Especially, since the sensor nodes are generally deployed in hazardous area and run unattended, the batteries used the nodes are irreplaceable. Therefore the efficient energy consumption becomes one of the most important issues to be concerned with. Generally the sensor network is composed of hundreds or thousands of sensor nodes, and a BS, or BSes. Each sensor node can collect environmental information by sensing unit and then it transmits sensed

[*] This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 58 – 67, 2006.

data to neighbor nodes or to a BS [1]. However, inter-sensor communication is normally within short transmission ranges due to energy and bandwidth limitations. Therefore, it is most likely that a route will consist of multiple wireless hops [3]. Below Figure 1 shows the architecture of wireless sensor network.

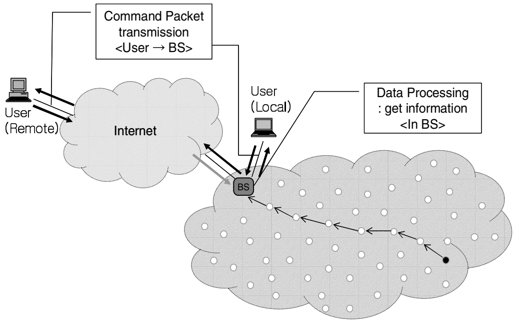


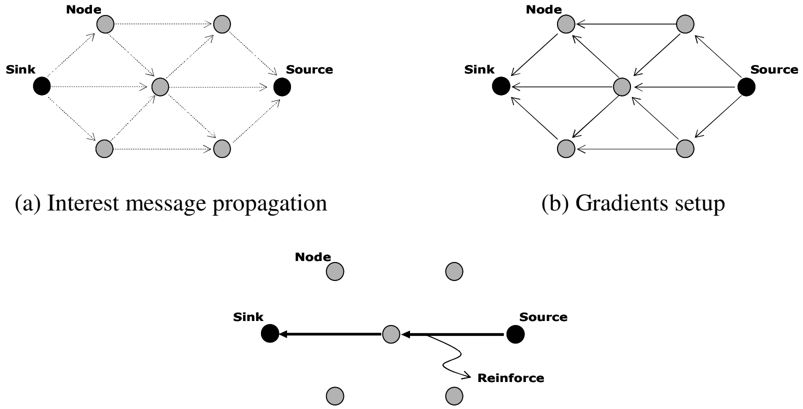
Fig. 1. Architecture of Wireless Sensor Network

A number of routing algorithms and protocols have been proposed for wireless sensor networks in recent years, with the goal of achieving more efficient and reliable data dissemination in wireless sensor networks [2,3]. The directed diffusion that is one of these routing protocols is a data-centric and data dissemination protocol to transmit a data between the BS and sensor nodes. This communication paradigm makes requests for collecting data of specific region as it propagates interest messages that were generated in the BS [5,6]. In this phase, the directed diffusion uses flooding to inject the interest message to all nodes in the sensor field. However, flooding may suffer from significant redundancy with many duplicated messages. To reduce the energy consumption due to the redundancy the BS can limit the flooding or transmission area to specific region [3,6-9]. To solve this problem, we propose a new fuzzy logic based routing method. The method limit the propagation or flooding area according to the result of the fuzzy logic that uses the factors like energy, density, and location information as its input.

The remainder of the paper is organized as follows: Section 2 introduces the directed diffusion as background knowledge. Section 3 briefly describes the routing algorithms for energy-efficiency and motivation of this work. Section 4 shows the details of the fuzzy logic based propagation limiting method. Section 5 reviews the simulation result. Finally, conclusion and future work is discussed in Section 6.

2 Directed Diffusion Overview

Generally, sensor network routing protocols can be classified into three types. These are flat network routing, hierarchical network routing, and location-based routing. In flat network routing, each sensor node typically plays the same role and collects data to perform the sensing task. This routing is an efficient way to reduce the amount of energy within the specific application, as performing data aggregation and elimination of redundant data [1-4]. Especially, the directed diffusion, one of these flat network routing is a data-centric routing algorithm, where the BS sends queries to specific regions and waits for data from the nodes in the specific regions [6]. An interest message is a query which specifies what a user wants. In order to create a query, an interest message is defined using a list of attribute-value pairs such as object type, interval, selected region, and etc. Figure 2 (a) shows the interest message propagation.



(a) Interest message propagation

(b) Gradients setup

(c) Data delivery along reinforced path

Fig. 2. The simplified process of directed diffusion

After the interest message is propagated throughout the sensor field, gradients are setup to draw data satisfying the query towards the requesting node. The gradient specifies an attribute value and a direction of data flow [6,7]. This process continues until the gradients are set up from the target regions back to the BS. Here is a Figure 2 (b) which shows setup of gradients. When a sensor node in the target region receives the message, it activates its sensors and begins to collect events. If the node discovers the event for query then the sensed data are returned in the reverse path of the interest message. The best paths are reinforced to prevent further flooding. We see from Figure 2 (c) that an example of the data delivery along reinforced path [3,4].

3 Related Work and Motivation

This section describes existing routing algorithms which consider energy-efficiency with or without localized query propagation. More researches related to energy-efficiency not introduced in this section can be found in [9-13].

The gradient-based routing (GBR) has proposed a slightly changed version of the directed diffusion [10]. The key idea is to record the number of hops when the interest message is propagated through the entire sensor network. As such, each node can discover the minimum number of hops to the BS, called the height of the node. The difference between a node's height and that of its neighbor is considered the gradient on that link. A packet is forwarded on a link with the largest gradient. These gradients indicate the goodness of the different possible next hops and are used to forward the sensed data to the BS [2,3]. This scheme strives to achieve an even distribution of the traffic throughout the whole network, which helps in balancing the load on sensor nodes and increases the network lifetime.

The geographical and energy aware routing (GEAR) [9] uses the geographic information while propagating the interest message to specific regions since the query data often includes geographic attributes. The key idea is to restrict the amount of flooding of the interest message in the directed diffusion by restricting the flooding area. As a result the GEAR can conserve more energy in the nodes than the directed diffusion [3,4]. In this algorithm, all the deployed nodes keep an estimated and learned costs to propagate the message through its neighbor nodes according to the location information, remaining energy level, that of neighbor nodes.

GBR upgrades the directed diffusion to setup gradient and reinforce path by hop count scheme. However, since this algorithm simply propagates the interest message to whole nodes in the network, it causes energy dissipation of nodes. This fact reaches a conclusion, since the BS needs not propagate the data throughout the network and it receives only a small amount of data from nodes in the target region, the use of flooding is unnecessary [1-3,6,9].

4 Propagation Limiting Method (PLM)

PLM uses the energy and density of sensor nodes to determine transmission area. The BS needs a threshold value to limit the transmission area. The threshold value is determined by a fuzzy logic system with the consideration of the energy and density. To archive a threshold value, the BS needs to store the information about the energy and density approximately. The BS creates a new interest message for queries, and the BS also needs two factors for this message creation. The energy and density become important factors to determine a threshold value that it limits propagation area of the interest message. Another factor is location information. If nodes desire to determine data propagation path through neighbor to neighbor, all nodes need to know their locations. We assume that the location of nodes may be available directly by communicating with a satellite using low power GPS and GPS cards [3,6,14].

4.1 Factors That Affect the Propagation Region

The energy is the most important and scares resource that should be considered first in sensor networks. Generally, sensor nodes are limited in power and irreplaceable since these nodes have limited capacity and are unattended. Therefore, it is important to consider the efficient energy dissipation scheme in the sensor networks [1]. If an interest message is propagated through the narrow propagation area that is comprised of nodes with low energy, then alternative paths to enable recovery of transmission failure will be not constructed [15]. So we have to decide the propagation area based on the energy level and density of the nodes. How this calculation is done is shown in the next section. Figure 3 illustrates the difference between the propagation area regarding the target region A (about 89% energy level and 92% density) and the propagation area regarding the target region B (about 46% energy level and 56% density).

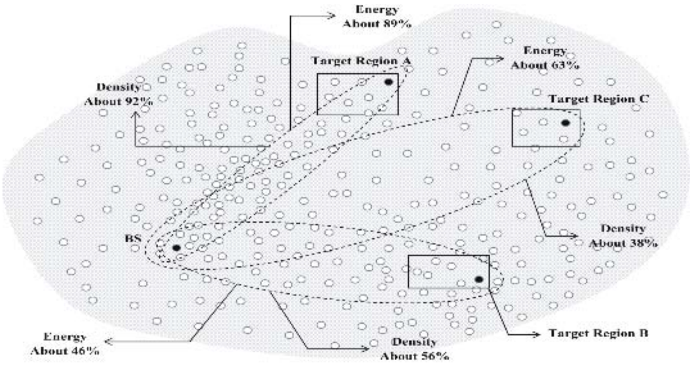


Fig. 3. Difference of transmission area based on remaining energy and density of networks

The number of nodes within the propagation area is used to indicate the network density [16]. Since communication between sensor nodes is normally done within short transmission ranges due to energy and bandwidth limitation, density is an important factor that is related to data reliability [3]. Therefore, the density is utilized to calculate the threshold value for determining propagation area in PLM. The propagation area of target region C, where the density is lower than the propagation area of target region A, is broader than region A.

4.2 Fuzzy Logic Based Threshold Value

In PLM, BS can limit propagation area of interest messages by adding a threshold value to the messages by fuzzy logic based selector. The selector determines a threshold value using input parameters of the energy and density stored in the BS. A propagation area low energy level covers a wide transmission scope. On the other hand, if density of nodes is high, although remaining energy is low, the propagation area is constructed on the narrow scope.

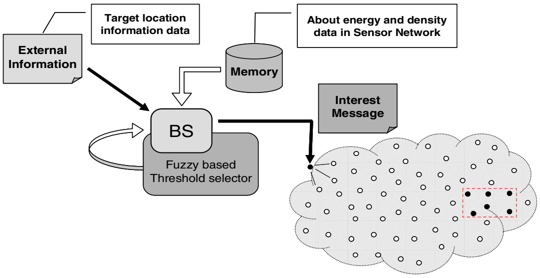

Target location information data
External Information
About energy and density data in Sensor Network
Memory
Interest Message
BS
Fuzzy based Threshold selector

Fig. 4. Architecture of the fuzzy-based selector

Figure 4 shows the architecture of the fuzzy-based selector for creating an interest message. As shown in the figure the fuzzy selector is located in BS and calculated the threshold value that is propagated to the deployed sensor nodes within the interest message.

Figure 5 illustrates the membership functions of two input parameters, energy and density, of the fuzzy logic. The labels in the fuzzy variables are presented as follows.

- ENERGY = {VERY SMALL, SMALL, HALF, MUCH, VERY MUCH}
- DENSITY = {VERY LOW, LOW, MEDIUM, HIGH, VERY HIGH}

The output parameter of the fuzzy logic is THRESHOLD = {VERY SMALL, SMALL, MEDIUM, LARGE, VERY LARGE}, which is represented by the membership functions as shown in Figure 5. The rules are created using the fuzzy system editor contained in the *MATLAB* Fuzzy Toolbox.

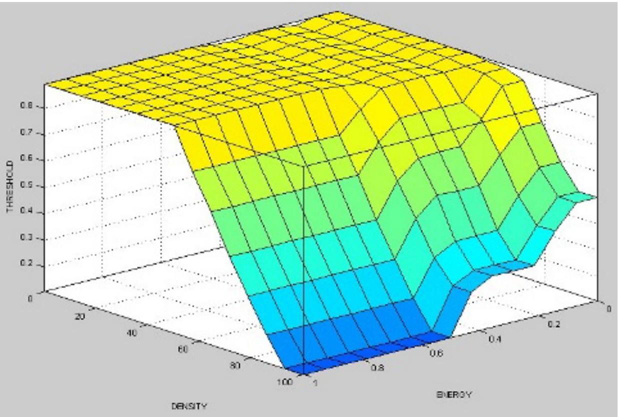


Fig. 5. Membership functions for input and output variables

If the ENERGY is SMALL and the DENSITY is VERY HIGH, then the threshold can take on a value below or above SMALL. Some of the rules are shown below.

R08: IF (ENERGY is **SMALL**) AND (DENSITY is **MEDIUM**)
THEN (THRESHOLD is **LARGE**)

R09: IF (ENERGY is **SMALL**) AND (DENSITY is **HIGH**)
THEN (THRESHOLD is **LARGE**)

R10: IF (ENERGY is **SMALL**) AND (DENSITY is **VERY_HIGH**)
THEN (THRESHOLD is **MEDIUM**)

4.3 Interest Message

When information on a specific region is requested, the BS can strategically select a subset of the network to sense the environment at a specific time. The BS determines a target region (Source) based on location information of deployed nodes, and it adds an optimal path *d* that is calculated by equation (1) to an interest message with the threshold value obtained by the fuzzy logic.

- BS's Location: B(b_x, b_y)
- Source's Location: S(s_x, s_y)

$$d = \sqrt{(s_x - b_x)^2 + (s_y - b_y)^2} \tag{1}$$

Figure 6 shows the nodes on straight line that is the optimal path between the BS and target region. In this case, the BS propagates interest messages along the nodes on straight line for reducing the communication cost in terms of energy. However, if the BS propagates the interest messages through the nodes on the straight line, it will not guarantee reliable transmissions.

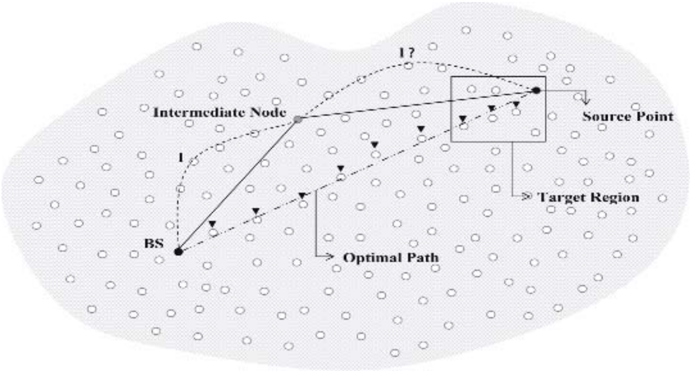


Fig. 6. The optimal path and intermediate node's real transmission distance between the BS and source point in target region

For this reason, the BS not only uses the nodes on the optimal path but also has to send the messages through the nodes near the path. The nodes to be included in the path are calculated based on the optimal path *d* and threshold value *t* (output of the fuzzy-based selector). That is, the BS creates the new threshold value *t'* which is a criterion value for determining the nodes that should be included within the propagation area. Equation (2) shows how the new threshold value *t'* is calculated. The equation is formulated considering the desired minimum and maximum values of *t'*. The *t'* becomes *d* and 2*d* when *t* is 0 and 1 respectively.

$$t' = d \times (t + 1) \tag{2}$$

Finally, the BS creates an interest message according to a threshold value, location information of nodes, and other information as explained in section 2.

4.4 Node Selection for Inclusion Within the Propagation Area

The interest messages created within the BS are forwarded toward the neighbor nodes. After an intermediate node receives the message and stores it in its interest entry, the node calculates distance using location information of the BS, itself, and a source point within the target region.

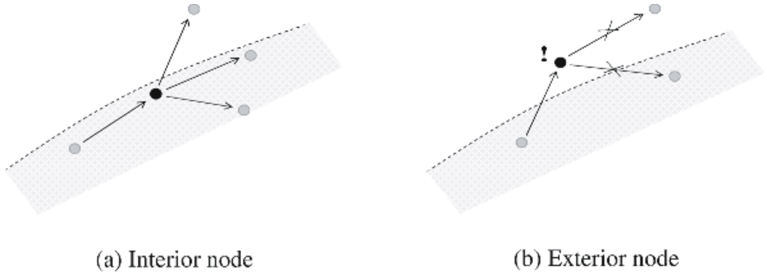
- Intermediate node's Location: N(n_x, n_y)

$$
\begin{aligned}
l &= \sqrt{(n_x - b_x)^2 + (n_y - b_y)^2} \\
l' &= \sqrt{(s_x - n_x)^2 + (s_y - n_y)^2} \\
d' &= l + l'
\end{aligned}
\tag{3}
$$

As described in equation (3), the distance value *l* and *l'* are calculated, and then a value *d'* is created by the sum of the two values. This value represents the real distance for transmitting an interest message from the BS to source point. The description of the real distance as mentioned above appears in Figure 6. After the intermediate node calculates *d'* as in (3), it compares against *t'* and then the node decides whether to itself in within the propagation area or not.

- IF *t'* ≤ *d'* THEN Interior Node
- IF *t'* > *d'* THEN Exterior Node

If the node is an interior node then it continues the propagation toward the target point as shown in Figure 7 (a), otherwise the propagation is terminated as shown in Figure 7 (b).


(a) Interior node

(b) Exterior node

Fig. 7. Nodes selection in propagation from BS to target node

After the nodes in the target region receive an interest message, then these nodes acquire requested data through their sensing unit. Thus acquired data is sent to the BS within a data message. BS uses the reinforce mechanism to select a high quality path for transmission of data message based on GBR [10] and GEAR [11]. The reinforce path is determined by the hop count and remaining energy. For the path creation, the interest message is required to record the hop count taken from BS. The data message is forwarded to a minimum hop neighbor node according to recorded interest message in the node. If the hop count of the neighbor node is equal to other nodes, the node selects a neighbor node by taking remaining energy into account. Finally, the BS receives the data messages and then reports the information to users.

5 Simulation Result

In this section, we evaluate the performance of fuzzy logic based propagation limiting method through simulations. We deployed sensor nodes in a rectangular area of

400×400 m^2. We assumed that each node's initial energy level is 1 J, the transmission cost for a message and reception cost for a message is 0.0001 J and 0.00005 J respectively. A node has a transmission range of 40 meters. We used two graphs for evaluating the PLM versus GBR.

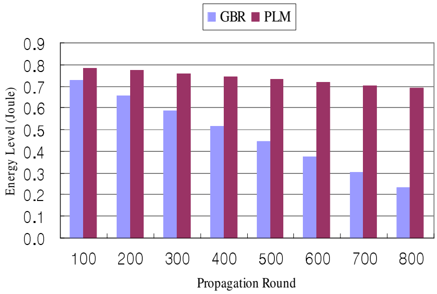


Fig. 8. Simulation result of energy consumption as propagation round

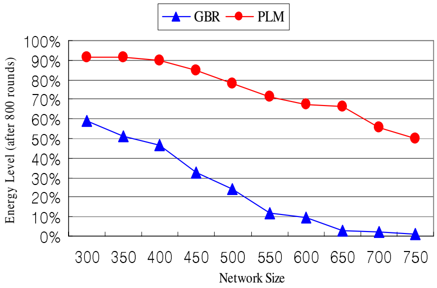


Fig. 9. Simulation result of energy consumption as network size

Figure 8 shows a result of the energy consumption against the interest message propagation round. As shown in the graph, the energy level of GBR is reduced approximately by 0.07 J each time the propagation round is increased by 100, whereas that of PLM is reduced approximately by only 0.015 J. The simulation results show that GBR consumes more energy than the fuzzy logic based PLM. Lifetime of sensor network using PLM is increased by about 2.94 times. Figure 9 illustrates simulation results regarding the network size in two different methods. Note that the energy in PLM maintains higher level independent of the network size.

6 Conclusion

In this paper, we present a fuzzy logic based propagation limiting method for energy-efficient data communication in wireless sensor networks. Our work is motivated by the directed diffusion protocol which propagates an interest message throughout the entire network to obtain the desired data without considering the efficient energy usage. The proposed PLM exploits the fuzzy logic system that uses energy, density, and location information as input in limiting the propagation area. The PLM consumes the limited energy more efficiently since only part of the nodes participates in the propagation process. Thus the nodes that do not participate in the propagation can save their energy, which results in the prolong network lifetime.

The security issues regarding the PLM are the next research topic.

References

1. Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A Survey on Sensor Networks. IEEE Commun. Mag. (2002) 102-114
2. Akkaya, K., Younis, M.: A Survey on Routing Protocols for Wireless Sensor Networks. Ad hoc Networks 3(3) (2004) 325-349

3. Al-Karaki, J. N., Kamal, A. E.: Routing techniques in wireless sensor networks: a survey. IEEE Wirel. Commun. 11(6) (2004) 6-28
4. Jiang, Q., Manivannan, D.: Routing Protocols for Sensor Networks. Proc. of CCNC (2004) 63-98
5. Estrin, D., Govindan, R., Heidemann, J., Kumar, S.: Next century challenges: Scalable coordination in sensor networks. Proc. of MobiCom (1999) 263-270
6. Intanagonwiwat, C., Govindan, R., Estrin, D., Heidemann, J., Silva, F.: Directed diffusion for wireless sensor networking. IEEE ACM T. Network 11(1) (2003) 2-16
7. Heideman, J., Silva, F., Intanagonwiwat, C., Govindan, R., Estrin, D., Ganesan, D.: Building Efficient Wireless Sensor Networks with Low-Level Naming. Proc. of SOSP (2001) 146-159
8. Ko, Y.B., Vaidya, N.H.: Location-Aided Routing (LAR) in Mobile Ad Hoc Networks. Wirel. Netw. 6(4) (2000) 307-321
9. Yu, Y., Govindan, R., Estrin, D.: Geographical and Energy Aware Routing: a recursive data dissemination protocol for wireless sensor networks. UCLA/CSD-TR-01-0023 (2001)
10. Schurgers, C., Srivastava, M. B.: Energy Efficient Routing in Wireless Sensor Networks. Proc. of MILCOM (2001) 357-361
11. Goldin, D., Song, M., Kutlu, A., Gao, H., Dave, H.: Georouting and Delta-gathering: Efficient Data Propagation Techniques for GeoSensor Networks. Proc. of GSN (2003)
12. Ye, F., Zhong, G., Lu, S., Zhang, L.: GRAdient Broadcast: A Robust Data Delivery Protocol for Large Scale Sensor Networks. ACM WINET 11(2) (2005)
13. Handziski, V., Köpke, A., Karl, H. Frank, C., Drytkiewicz, W.: Improving the Energy Efficiency of Directed Diffusion Using Passive Clustering. Proc. of EWSN (2004) 172-187
14. Xu, Y., Heidemann, J., Estrin, D.: Geography-informed Energy Conservation for Ad-hoc Routing. Proc. of MobiCom (2001) 70-84
15. Ganesan, D., Govindan, R., Shenker, S., Estrin, D.: Highly-resilient, energy-efficient multipath routing in wireless sensor networks. *Mobile Computing and Comm. Rev.* 4(5) (2001)
16. Bulusu, N., Estrin, D., Girod, L., Heidemann, J.: Scalable Coordination for wireless sensor networks: Self-Configuring Localization Systems. Proc. of ISCTA (2001)

Content Delivery with Spatial Caching Scheme in Mobile Wireless Networks

Backhyun Kim and Iksoo Kim

Department of Information and Telecommunication Engineering, Univ. of Incheon,
177 Towha-Dong, Nam-Ku, Incheon, Korea
{hidesky24, iskim}@incheon.ac.kr

Abstract. Future wireless/mobile system will be served streaming service to support various applications. In this environment, users can access all the required information whenever and wherever they may be. In this paper, we proposed a new scheme that minimizes the network bandwidth consumption and the service blocking rate in mobile network. Our proposed scheme consists of location estimation and caching strategy. To estimate node's current location, we use hexagonal cellular based plane and cascading dynamic address scheme which make mobile equipments, mobile router (MR) and mobile node (MN), calculate the distance from sender and estimate the existence possibility of the alternative route toward sender. The service blocking can be minimized by using caching strategy. In evaluation, we examined our proposed scheme in the view of the total network bandwidth consumption and the connecting probability as well as the impact of each cache capacity. From simulation results, we confirm that the proposed scheme offers substantially better performance.

1 Introduction

In ubiquitous computing, individual users utilize several electronic platforms through which they can access all the required information whenever and wherever they may be. This application should be operated in mobile environment formed in wireless LAN and even on the human body called personal area networks (PAN). Mobile nodes (MNs) can communicate with their intended destinations that can be either in fixed or in mobile environment. This communication environment is Internet-based Mobile Ad hoc Networks (IMANETS) that consists of the wired Internet and Mobile Ad hoc Networks [1]. Under this scheme, MN can be linked directly when it wants to connect to Internet through access point (AP) of wireless LAN. When MN is located out of AP's transmission range, the connection to Internet can be done with ad hoc protocol.

This paper presents a new data delivery technique that adopts a spatial caching strategy to reduce the network traffic and the service latency, and an dynamic address allocation scheme that can simplify the route establishment and maintenance. In this paper, we use short distance vector algorithm based on the proactive and the flat topology routing protocol in MANET. Delivery tree construction sequence is started by a MN which wants to receive multimedia data from a source node or another MN.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 68 – 77, 2006.

The transmission sphere of MN can be divided into six regions at most similar to hexagonal plane of mobile cellular network. Basic idea of our address allocation scheme is to separate node identity and node address. Node address indicates node's current location and is allotted by its parent node with a cascading address scheme. Under the proposed scheme, one of MNs is elected as address allocator (AA). We assume that AA is access point (AP) in IMANET. The transmission area of AA can be divided into six regions at most similar to hexagonal structure of mobile cellular network. MNs in each region have same address assigned by AA and one of them becomes a root node for each delivery tree. Thus six root nodes can make six different delivery trees. The address for MN 2-hops away from AA is assigned by its root node and becomes its root node address followed by newly generated region address. Therefore, the address for MN n hops away from AA is generated by MN n-1 hops away from AA on delivery route tree. MNs in the same region are classified by node identifications as IP address, MAC address, etc. With proposed address scheme, as the address of destination node indicates the entire multi-hop path from a source to destination, each MN maintains only routing information for MNs 1-hop away from itself to keep up-to-date route.

To support seamless streaming service, we adopt caching system into MN. All nodes on delivery tree cache and forward the streaming data. If a request node can connect to the previously constructed delivery tree, the request item can be served with the data cached in MNs along the delivery tree. We assume that multimedia content consists of equal-sized segments and MN caches the same content by a segment. The data cached in MN n-hops away from a source is n'th segment of the content. If MN moves along the path delivering the same content, as it caches a series of segments, low service latency and blocking rate can be achieved. A cache replacement policy is based on a popularity and distance of cached item to improve the accessibility as multimedia data is time-sensitive.

The remainder of the paper is organized as follows: The next section we summarize previous work. Section 3 describes a dynamic address allocation scheme and a mobility modeling with random-walk in hexagonal cellular architecture plane. Proposed real-time streaming service using caching technology is introduced in section 4. In section 5, we present the simulations and analysis of the results. Finally, we give out conclusion in section 6.

2 Previous Work

MANET operates in distributed manner as there is no control node such as AP and BN. If MN wants to communicate with another one, it should find the proper path between itself and that one. The routing protocols of MANET are split into two categories based on the routing information update mechanism; proactive and reactive routing protocols. In proactive [2], [3] called table-driven routing protocol, every node maintains the entire network topology in the form of routing tables by exchanging routing information periodically and this information is flooded over the whole network. Though this method can achieve up-to-date route information, it consumes much more network bandwidth by generating unnecessary traffics. Reactive [4], [5] is called on-demand routing as it executes path finding procedure only when MN wants to communicate with another one. So in reactive routing, every node does not need to maintain the whole network topology while proactive routing protocols do. This method can reduce the network traffic, not to flood routing information periodically, and thus increase network throughput. But since it should set up the route before sending a request data, service will be delayed until the route setup sequence will be completed. MANET can use of either a flat topology or a hierarchical topology for routing. In flat topology, address for MN is globally unique and routing information for updating and maintaining network topology is flooded over whole network. Hierarchical mechanism makes an entire flat topology a logically hierarchical structure called zone limited within a particular geographical region.

Dynamic address allocation can simplify routing procedure. Thoppian et al. [6] propose dynamic address assignment that allots a unique IP address to a new node joining in MANET. In this scheme, each MN has some IP address block. When a new node (requester) join a MANET, one of the existing MANET nodes (allocator) within communication range of the requester allots the second half of the addresses from its free_ip set to the requester. Though this method guarantees unique IP address assignment under a variety of network conditions, it cannot reduce routing overhead as each node has the same length of address, i.e. 4 bytes in IPv4. Eriksson et al. [7] propose a variable length of dynamic addressing scheme based on a hierarchical binary tree structure of proactive distance vector routing. This scheme separates node identity from node address that indicates the node's current location in the network. If there are n nodes, address length is $\log_2 n$ and average routing table sizes are less than $2\log_2 n$. Node lookup information is distributed in the network as node lookup to find the current address of a node is done by hash function. As address length is proportional to the number of nodes in the network, it produces routing overhead in a dense network and data will be taken along a longer path instead of the shortest route. Chen and Nahrstedt [8] propose address compression scheme to reduce routing overhead with overlay multicast in MANET. This method also separates node identifier from node address called index. A node's index is determined by the application server when it joins the multicast group. Node lookup is done by the application server and then this information is recorded in each node's address lookup table. As this scheme assigns unique index to each node, it is not a proper scheme in dense network.

Cao et al. [9] introduce a cooperative caching scheme in which multiple nodes share and coordinate cached data. Router node caches the data when it finds that the data is frequently accessed except all requests for the data are from the same node. A node caches the path to the caching node only when it satisfies the closeness condition that is defined as a function of distance to the data source, route stability, its distance to caching node, and the data update rate. This can reduce network traffic and even provide service if node can not connect to the server in the meantime. But nodes storing popular items are faced with heavier traffic than others. Lim et al. [10] propose an aggregate caching mechanism that same data items are cached at least Γ hops apart, where Γ is a system parameter. To increase accessibility, they try to cache as many data items as possible as it is meaningless to reduce access latency when a set of nodes is isolated from other nodes or AP. In this scheme, when a number of data item will be requested, accessibility will be decreased due to the limited size of cache. Also there is unbalanced load among caching nodes storing the same data items as the amount of data cached among them is different.

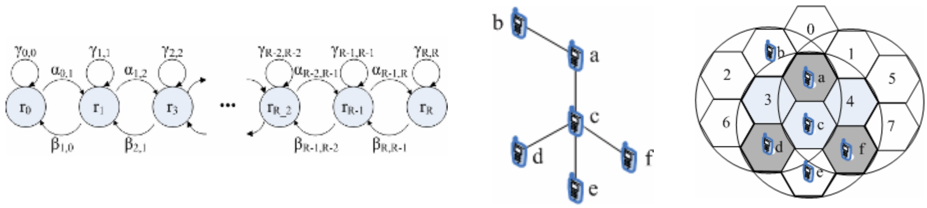
3 Address Allocation

In MANET the route can be frequently changed due to the node's movement. Hexagonal plane makes the analysis of mobility and connectivity of mobile nodes easy. To evaluate the mobility of MN, there are some proposals [11], [12], [13] over a hexagonal cellular networks structure. In these papers, each cell has same size and is managed under a fixed control node. Each ring r_i, where $i \geq 1$, is composed of $6i$ cells. The ring r_0 is called the center cell or central ring comprising a single cell. The ring r_{i-1} is surrounded by ring r_i, where i is the distance calculated by means of the number of cells from the center cell r_0. Therefore the number of cells up to ring r_R in hexagonal cells is $\sum_{i=1}^{R} 6i+1$. MN can remain within its current cell or move to another adjacent cell, i.e. neighboring cells if there are no cells between two cells mentioned above. If p is the transition probability of MN, the remaining probability is $1 - p$. In hexagonal cellular model, there are 6 neighboring cells at each one and MNs have equal probability that they move to one of the adjacent cells and their probability is equal to 1/6. But the transition probabilities to cell $i - 1$, i or $i + 1$ are changed to $p/6$, $p/3$ and $p/2$, respectively.

To make user mobility model, we use random-walk mobility model. This model is suitable when MNs move within a geographically limited area with frequently change in their moving directions. To evaluate the mobility of MN, we use Markov chain model shown in Fig. 1(a). The transition probability $\alpha_{i,\, i+1}$ represents the probability that MN moves from cell i to cell $i + 1$, i.e. moves away from the center cell. $\beta_{i,\, i-1}$ represents the probability that MN moves from cell i to cell i - 1, i.e. moves toward center cell. $\gamma_{i,\, i}$ represents the probability that MN remains its current ring. If MN locates in the ring r_i, the probabilities that a movement will result in an increase or decrease in the distance from the center cell or remain at its current ring, denoted by $p^+(i)$, $p^-(i)$ and $p^0(i)$, respectively, are

$$p^{+}(i) = (\frac{1}{3}+\frac{1}{6i})p, \quad p^{-}(i) = (\frac{1}{3}-\frac{1}{6i})p, \quad p^{0}(i) = \frac{1}{3}p \tag{1}$$

But, since MN can move to only outer side of its location when it locates in the center ring r_0, the total transition probability p is equal to $p^+(0)$, i.e. $\alpha_{0,1}$, and the remain probability 1-p becomes 1- $p^+(0)$, i.e. $\gamma_{0,0}$. It is hard to identify MN's current location as there is no coordinator node or infrastructure to verify location such as GPS in MANET. Fig. 1(b) shows the communication links among MNs <a, b, c, d, e, f> with short distance vector routing protocol. Let MN a be the source node and MN <b, d, e, f> be the requesters. MN <b, c> become the child nodes of MN a, and MN <d, e, f> become the child nodes of MN c. MN c becomes switch node to forward the data requested by MN <d, e, f>. MN <d, e, f> have the sibling relationship since they have same distance from MN a as a source node. Fig. 1(c) shows the link-based current location when communication links are mapped into hexagonal cellular plane. Let $N(n)$ be link table consisted of the adjacent nodes called neighbors that can directly communicate with MN n. From Fig. 1(c), $N(a)$, $N(b)$, $N(c)$, $N(d)$, $N(e)$ and $N(f)$ are <b, c>, <a >, <a, d, e, f>, <c, e>, <c, d, f> and <c, e>, respectively. Each node periodically broadcasts link table which is consisted of the adjacent nodes only. When node receives link tables of neighbors, it can estimate the current locations of the adjacent nodes. For example, MN c knows the existence of MN d, e and f. From received link tables from MN d, e and f, MN c can perceive that MN d and f do not know the existence of each other but MN e knows the existence of MN d and f. Thus MN d and f locate out of each transmission range and MN e locates at between them.



(a) Markov chain model

(b) Communication links

(c) Mapping into hexagonal plane

Fig. 1. Location estimation plane based on hexagonal cellular architecture

We assume that AP or a source MN is located in cell r_0, and it splits its transmission region into equal-sized 6 hexagonal cells, and transmits the data to each cell independently and identically. Delivery tree can be constructed only when two connecting MNs are located in different and adjacent cells. If more than two MNs are located in same cell, basically they cannot connect each others except they cannot reach their parent MN in which locates a different cell. Let $H(n)$ be the number of hops away from center cell based on short distance vector routing algorithm. $H(n)$ of MNs in cell <a>, <0, 1, 3, 4, b, c> and <2, 5, 6, 7, d, e, f> are 0, 1 and 2, respectively, where $H(n) = 0$ means itself. In hexagonal plane, MN can move only 6-directions from its current location. Let $P_r(c)$ and $P_e(c)$ be the remaining probability of MN in cell c and the existence probability of the route from BN to cell c, respectively. $P_r(c)$ can be calculated as the function of the probabilities that MNs in cell c will remain and the MNs in 6 adjacent cells of cell c will move to cell c. Let $p(x, y)$ be the transition probability that MN y in a neighboring cells of cell x moves to cell x. Therefore, $P_r(c)$ and $P_e(c)$ in cell c which is h hop away from BN, are calculated using the following equations:

$$P_r(c) = 1 - \left(\prod_{i=1}^{n} p(c,i) \times \prod_{j=1}^{a} \left(\prod_{i=1}^{n} p(j,i) \right) \right), \qquad P_h(c) = \prod_{i=1}^{h-1} P_r(i) \tag{2}$$

where n and a are the number of MNs and neighboring cells in each cell, respectively. After finishing route construction scheme described in next section, MNs may be connected with MNs in 2 adjacent cells at most. The route will be broken if there are no MNs in cell c due to the MNs' movement. But the route can remain connecting if there is a detour via adjacent cells along the data path. Let $P_d(c)$ be the existence probability of the detour alternating with cell c. $P_h(c)$ in Eq. (2) can be modified as the following equation:

$$P_h(c) = \prod_{i=1}^{h-1} \left((1 - P_r(i)) P_d(i) + P_r(i) \right) \tag{3}$$

4 Spatial Cache Scheme

In this paper, nodes can communicate with each others with tree-based short distance vector routing protocol. This model transmits data by forwarding to one of its neighbors which is closer to the destination node and node's transmission range is fixed as R. These data therefore propagate from source to the destination by hop from one node to another until they arrive at the destination node. We assume that all nodes can communicate to any other nodes in the network. For this, we make a simulation model satisfying the condition mentioned in [14]. To estimate connectivity, we use following assumptions: 1) Nodes in the networks are placed in a disc of unit area. 2) The location of each node can be modeled as Poisson random process. 3) Each node can communicate with transmission range R at a power level so as to cover a unit area as a rectangular planer of which both height and width are D.

We proposed a dynamic address allocation scheme to estimate MN's current location described in previous section. The proposed scheme separates node identity and node address. Node address indicates the overall route from AP to node and represents a cell in which node locates. From Fig. 1(c), as one cell may have nodes more than one, it is needed the mechanism to identify among them. For this, we use IP address as node identifier. Since MNs in ring r_1 locate at one of 6 cells, 3 bits address is sufficient to identify them. A cell in ring r_1 may have 6 cells that are consisted of 1 parent cell, 2 sibling cells and 3 child cells according to their locations on the delivery tree. Therefore, the other cells locating at r_i, where $i \neq 0$, can have 3 child cells at most and they can be identified with 2 bits. In Fig. 1(c), source node locates at cell a. Node b and node c are located at ring r_1 and their addresses become 001 and 010 as they can not directly communicate with each other. Node c has 3 child nodes <d, e, f> that locate at cells named the same notations. To identify child nodes, node c allots the address 01, 10 and 11 to them in order. As a result the address lengths of nodes <d, e, f> become 010-01, 010-10 and 010-11, respectively. The address of a MN at i hops away from AP can be represented by a cascade addressing algorithm which makes new address with a received address, indicating the address till i-1 hop, followed by its current location address.

To support streaming the data, the time to keep up-to-date route information can be minimized but it is hard to implement. So we adopt cache system into mobile nodes to reduce the impact of delay to change the route and the amount of consumed network bandwidth to deliver multimedia contents [14]. Let MN$_i$ be the MN at i hops away from AP. Each MN can store the identical item by $S_c = I_t R_{CBR}$ bits, where R_{CBR} and I_t are constant bit rate (CBR) for streaming the data and the predefined time interval, respectively. AP can service the multimedia content of which size is nI_t, i.e. multimedia content consists of n blocks and each block has the same size I_t. Node caches only i'th block if it is at a cell i hops away from AP. During the time interval I_t the total amount of consumed network bandwidth for delivering the data to MN$_\mathrm{i}$ is $i{\cdot}S_c$ and the total amount of cached data among MNs is S_c. Assume that MN$_\mathrm{i}$ started caching the incoming data at $\mathrm{t}_{\mathrm{scache}}$ and finished at $\mathrm{t}_{\mathrm{fcache}}$. When $\mathrm{t} > \mathrm{t}_{\mathrm{fcache}}$, MN$_\mathrm{j}$ requests the same item cached in MN$_\mathrm{i}$ to AP. If request packet (REQ) from MN$_\mathrm{j}$, $j < i$, travels via MN$_\mathrm{i}$ to AP, the route for MN$_\mathrm{j}$ can be simply constructed by attaching to preexisting route as child node of MN$_\mathrm{i}$.

If $j > i$, REQ of MN_j travels via MNs <MN_{j-1}, …, MN_{i+1}, MN_i, MN_{i-1}, …, MN_1, AP>. When MN_i receives this REQ, first it verifies whether REQ is a request for the cached data at its own cache or not. If requests for cached data, MN_i sets its forward cache data bit (fwC) to one. And MN_i adds its hop count i to the received REQ to indicate how much data have been cached among MNs over a delivery route, and then broadcasts REQ to the adjacent MNs. This modified REQ is delivered to MN_{i-1} as MN_i has already known the route to AP. MN_{i-1} verifies it, sets its fwC, add its hop count i-1, and broadcasts it. This procedure is repeatedly done by MN-by-MN till reaching AP. When AP receives this REQ, it sends back an acknowledgement packet (ACK) to MN_j if it is valid request. Since there are i cached blocks, iI_t, among MNs over delivery route, AP does not send first $iI(t)$ blocks and will send the rest parts of the data $(n-i)I_t$ after iI_t time.

The ACK traverses the reverse route of REQ. When MN_1 receives ACK from AP, it sends ACK to MN_2 and starts streaming the request data which has already been stored in its own cache at time t_s. The duration of sending cached data is I_t and the amount of delivered data is $[0, I_t]$, to denote the data block from time 0 to time I_t. Both AP and MN_2 can calculate the starting time of sending the request data by perceiving the radio signals from MN_1, and synchronize their timer with MN_1's timer. At time $t_s + I_t$, MN_2 starts sending 2'nd I_t block as long as $[I_t, 2I_t]$ for I_t. This procedure is repeated by MN-by-MN until MN_i finishes sending the cached data. After time $t_s + iI_t$, since the rest parts of multimedia content $[iI_t, nI_t]$ have not been cached among MNs over delivery route, AP starts sending these blocks. So the total amount of consumed network bandwidth, B_t, to send data as long as nI_t can be calculated as the sum of B_c, B_f and B_r, where B_c, B_f and B_r are the total amount of consumed network bandwidth for sending cached data among MNs, for forwarding the cached data among MNs behind MN_i, and for delivering the rest parts of data $(n-i)I_t$, respectively. B_t can be calculated as follows.

$$B_t = B_c + B_f + B_r = I_t R_{CBR}\left(\sum_{x=1}^{i-1} x + (j-i)(j-1) + j(n-i)\right) \tag{4}$$

Let B_{nc} be the amount of delivered data when not using caching. From the result of Eq. (4), the amount of reduced bandwidth for sending a request data compared with the case not using cache is

$$B_{nc} - B_t = \left[\sum_{x=1}^{i} xI_t R_{CBR}\right] \tag{5}$$

The REQ of MN_j at $j \le i$ travels along the path <MN_{j-1}, …, MN_1, AP>. In different from the case $j > i$, j will be connected with a MN at j – 1. The request MN_j can be located with another forwarding MN_j in the same cell j at the same time. When MN_{j-1} receives REQ of MN_j, it sends REQ not only to MN_{j-2} but also to forwarding MN_j to delivery the data cached among MNs from MN_j to MN_i. If MN_{j-1} receives ACK generated from AP, it sends reply packet (REP) to one of its child MNs that have sent ACK to confirm the procedure for sending cached data if their hop counts are $j \le i$. From Eq. (4), the total amount of consumed network bandwidth can be achieved as follow;

$$B_t = I_t R_{CBR}\left(\sum_{x=1}^{j-1} x + \sum_{y=1}^{i-j} y + j(n-i+1)\right) \tag{6}$$

The amount of transmission for sending cached data among MNs is much less than that of AP does. From Eq. (7), j should be less than $(i+1)/2$ because the amount of transmission data from MN_y at $y > j$ is much larger than the amount of transmission data from MN_x locating $x < j$ if MN_j locates at a cell closer to AP.

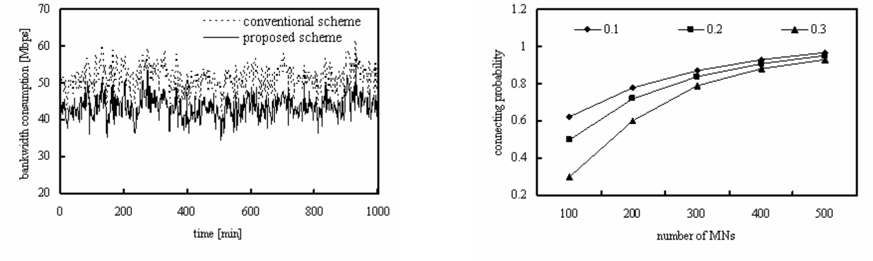
$$I_t R_{CBR}\left(\sum_{x=1}^{j-1} x + \sum_{y=1}^{i-j} y + j\right) < ijI_t R_{CBR} \tag{7}$$

The cached data should be deleted due to the cache capacity. For this, we use cache replacement policy based on a popularity and distance of the cached data. If a node moves along the path from cell i to cell i+n, it has cached the data $[iI_t, (i+n)I_t]$. The popularity is measured by the access frequency of the cached data. When more than one cached data have the same distance, a victim is selected with their popularity. From the cut off zipf's like distribution [15], as the request frequencies of unpopular items are almost same, the distance is used to remove these items. The distance is measured by the number of hops away from the current location. Thus a cached data with the highest distance is selected as a victim.

5 Simulation and Analysis

In this section, we show simulation results to demonstrate the benefit of proposed mobile ad hoc network with the caching and the location estimation mechanism to support robust streaming service, and analyzes on the results of performance using it. Let the size of cache at each MN be equal to $3I_t$ and I_t be 1 minute long. AP serves the multimedia contents of which sizes are 30 minutes long. We assume that simulation network is created within a 2000m x 2000m space with 400 MNs that are homogeneous and energy-constrained. The transmission range of nodes is selected from uniform distribution from 100m to 200m but we set this 150m default. Basically, the proportion of MNs to communicate with BN is 10% and the transition probability of MNs p is 0.3 and the service request rate λ follows Poisson distribution.

Fig. 2(a) shows the comparison between conventional scheme and proposed scheme in the view of the total network bandwidth consumption in entire network. The X-axis shows the time after simulation starts while the Y-axis shows the total network bandwidth consumption in entire MANET in the unit of bitrate as 128 kbps. Form the



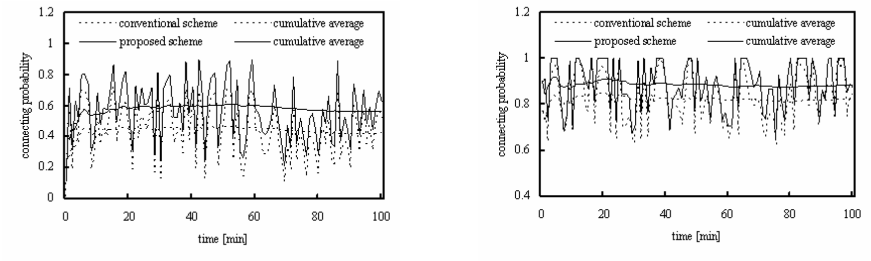
(a) Network bandwidth consumption

(b) Connecting probabilities as the function of the various transition probability p

Fig. 2. Simulation results

result, proposed scheme saves the total network bandwidth consumption compared to the conventional scheme all the time. Fig. 2(b) shows the variation in the connecting probability as the function of the various transition probability *p* ranged from 0.1 to 0.3 under the proposed scheme. The X-axis shows the number of MNs while the Y-axis shows the connecting probability in entire MNs. This result is derived from calculating the number of blocking route after all routes for ReqMNs are successfully established. The result shows that the higher connecting probability can be achieve as the transition probability is lower. Also, if the number of MNs increases, the connecting probability also increases as the residence probability of MNs in each cell increases. Therefore the connecting probability is inversely proportional to the transition probability.

Fig. 3 shows the variation in the network bandwidth consumption where the number of MNs is 200 and 400, in conventional scheme and proposed scheme. The X-axis shows the time after simulation starts while the Y-axis shows the connecting probability. The connecting probability means the probability that the ReqMN is continuously served even if the route to AP is broken due to the movement of MNs over delivery tree. From the result, proposed scheme reduces the duration of service blocking compared to the conventional scheme all the time, where cumulative average indicates the mean connecting probability from time 0 to time t. If MN locates at a cell *c* which is more than 10 hops away from AP, the amount of cached data among MNs over its delivery tree is 30 minutes long. Thus AP does not need to send the requested data because the length of content is 30 minutes long. If the pass is broken at hop 1, i.e. pass broken between AP and parent MN at 1 hop away from AP, since ReqMN can receive the request data from its parents which cached data [0, 10I_t], a pass break does not influence the ReqMN. As this reason, the proposed scheme can achieve better connecting probability than conventional scheme.



(a) MNs = 200 (b) MNs = 400

Fig. 3. Connecting probability according to the number of mobile nodes in the network

6 Conclusion

In this paper, we proposed a content delivery scheme to achieve the robust streaming of multimedia contents, consisting of region-based dynamic address allocation scheme and caching strategy. The former makes MNs estimate their current locations determined by a hop-count and a relationship among adjacent nodes in hexagonal cellular architecture, without any infrastructure to support location information. MN can split their outer areas into multiple cells and establish alternative route by selecting another MN locating at inner cell when the route is broken. Proposed address scheme make MN calculate the distance to AP and estimate the existence possibility of the alternative route toward AP. The latter can reduce the network traffic and the service blocking rate due to the pass breaks. Spatial caching scheme does not try to cache the entire data on an identical item but try to cache only one data block of item according to node's current location on delivery route. Thus it can reduce the load of caching nodes and the heavier traffic near caching nodes. In evaluation, we examined our proposed scheme in the view of the total network bandwidth consumption and the connecting probability as well as the impact of each cache capacity. The simulation results indicate that the cache size is the critical performance factor in order to minimize the network bandwidth consumption. In our future works, we will consider more different mobility models and location estimation schemes. Also, distributed caching mechanism and multicast delivery which can support real-time service are currently underway.

References

1. M.S. Corson, J.P. Maker, and J.H. Cernicione, "Internet-based mobile ad hoc networking," IEEE Internet Computing, Vol.3, no.4, pp. 63-70, 1999
2. J.T-C. Tsai, T. Chen, and M. Gerla, "QoS routing performance in mulithop, multimedia wireless network," Proc. IEEE ICUPC'97, 1997
3. J. S. Ho and I. F. Akyildiz, "Mobile user location update and paging under delay constraints," ACM-Baltzer J. Wireless Network, Vol. 1, pp.413-425, Dec. 1995
4. I.F. AKyildiz and W. Wang, "A dynamic location management scheme for next-generation multitier PCS systems," IEEE Trans. Wireless Commun., Vol.1, no.1, pp.178-189, Jan. 2002
5. S. Pack and Y. Choi, "A study of performance of hierarchical mobile IPv6 in IP-based cellular networks," IEICE Trans. Commun., Vol. E87-B, no.3, pp.462-469, March 2004
6. M.R. Thoppian and R. Prakash, " A distributed protocol for dynamic address assignment in mobile ad hoc networks," IEEE Trans. on mobile computing, Vol.5, No.1, pp.4-19, Jan. 2006
7. J. Eriksson, M. Faloutsos, and S. Krishnamurthy, "Scalable ad hoc routing: the case for dynamic addressing," In Proc. Of IEEE Infocom 2004, 2004
8. K. Chen and K. Nahrstedt, "Effective location-guided overlay multicast in mobile ad hoc networks," International Journal of Wireless and Mobile Computing (IJWMC), Special Issue on Group Communications in AD Hoc Networks, Vol. 3, 2005
9. G. Cao, L. Yin, and C.R. Das, "Cooperative cache-based data access in ad hoc networks," IEEE computer, Vol.37, No.2, pp.32-39, 2004
10. S. Lim, W.C. Lee, G. Cao and C.R. Das, " Performance comparison of cache invalidation strategies for internet-based mobile ad hoc networks," IEEE international conference on mobile ad hoc and sensor systems (MASS), pp.104-113, Oct. 2004
11. J. S. Ho and I. F. Akyildiz, "Mobile user location update and paging under delay constraints," ACM-Baltzer J. Wireless Network, Vol. 1, pp. 413-425, Dec. 1995
12. I.F. AKyildiz and W. Wang, "A dynamic location management scheme for next-generation multitier PCS systems," IEEE Trans. Wireless Commun., Vol.1, no.1, pp.178-189, Jan. 2002
13. S. Pack and Y. Choi, "A study of performance of hierarchical mobile IPv6 in IP-based cellular networks," IEICE Trans. Commun., Vol. E87-B, no.3, pp.462-469, March 2004
14. P. Gupta and P.R. Kumar, "Critical Power for Asymptotic Connectivity in Wireless Networks," A Volume in Honour of W.H. Fleming in Stochastic Analysis, Control, Optimization and Applications, 1998
15. L. Breslau, P.Cao, L, Fan, G. Phillips, and S. Shenker, "Web Caching and Zipf-like Distributions: Evidence and Implications", Proc. of the Conf. on Computer Commun (IEEE Infocom), New York, Mar 1999

Higher Education Web Information System Usage Analysis with a Data Webhouse

Carla Teixeira Lopes[1] and Gabriel David[2]

[1] ESTSP/FEUP, Portugal
carla.lopes@fe.up.pt
[2] INESC-Porto/FEUP, Portugal
gtd@fe.up.pt

Abstract. Usage analysis of a Web Information System is a valuable help to predict user needs, to assess system's impact and to guide to its improvement. This is usually done analysing clickstreams, a low-level approach, with huge amounts of data that calls for data warehouse techniques. This paper presents a dimensional model to monitor user behaviour in Higher Education Web Information Systems and an architecture for the extraction, transformation and load process. These have been applied in the development of a data warehouse to monitor the use of SIGARRA, the University of Porto's Higher Education Web Information System. The efficiency and effectiveness of this monitorization method were confirmed by the knowledge extracted from a 3 month period analysis. A brief description of the main results and recommendations are also described.

1 Introduction

The Web is growing in the number of users [12], usage rate [12] and complexity of its sites [5]. The use of this medium as an access interface to organizational Information Systems (IS) and their applications is also frequent. As the experience and expectation of users increases, the need to know and meet user demands becomes more pertinent. Monitoring users' behaviour helps to know their needs and allows system adaptation based on their previous behaviours [15]. Besides system adaptation, it also: supports the evaluation of the system against its initial specifications and goals, enables the development of personalization strategies [1, 4, 6], helps increase system's performance [6, 7], supports marketing decisions [3], helps detect business opportunities that otherwise could remain unnoticed [10] and may contribute to increase the system's security [4, 14].

Monitoring the use of Web Information Systems (WIS) involves analysing clickstreams, a data source that aggregates information about all user actions in a website. Log file analyzers, applications that extract data directly from log files and generate several kinds of statistics, are one of the most adopted solutions to monitor WIS usage [11]. However, with this technique it's hard, if not impossible, to obtain the level of analysis that other techniques allow. Log file analyzers lack the ability to integrate and correlate information from

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 78–87, 2006.

different sources. They can't, for example, correlate the number of accesses from a student to the web site with the program he is enrolled into. An alternative with more analytic potential, suitable to process large quantities of data (as happens with clickstream data), involves using a data webhouse, this is, a data warehouse that stores clickstreams and other contextual in order to understand user behaviour [8].

In Section 2 a dimensional model suitable to monitor Higher Education Web Information System (HEWIS) is presented. This has been the model used in the data webhouse to monitor the usage of the University of Porto's (UPorto) HEWIS. The architecture and a description of the processes involved in the extraction, transformation and load (ETL) are presented in Section 3.1. In the following section, some of the main results and in Section 3.3 some recommendations are presented. Conclusions and lines of future work are presented in the last section.

2 Dimensional Model

Considering the HEWIS scenario, a dimensional model to monitor this specific type of WIS usage has been defined. This process has begun with context analysis, followed by the establishment of the granularity, the definition of the relevant dimensions and facts identification.

2.1 Granularity

Not forgetting that dimensional models should be developed with the most atomic information [9], when the business process is associated with very large quantities of information, it is crucial to choose a granularity that is meaningful to the user and that, simultaneously, adds value to the organization's knowledge. Since the main goal of the present data warehouse is the analysis of user behaviour it has been decided to implement a granularity of web pages (see Figure 1) and web sessions (see Figure 2). The web page grain will allow answering questions related to user actions inside sessions, which is not possible with just a session fact table. The web session grain allows greater performance on questions related to WIS sessions.

2.2 Dimensions

The model has 12 dimensions that will be described next. The Academic Date, User, Page, Session Type and Institution dimensions are specific to the higher education context.

Access Date. The Access Date dimension stores information about the day of the civil calendar day in which the request was made. It only has one hierarchy with four levels: Year, Quarter, Month and Day.

Time of Day. To avoid the size of a dimension that saves the time of day for each day of the civil calendar, it has been decided to split time into a new

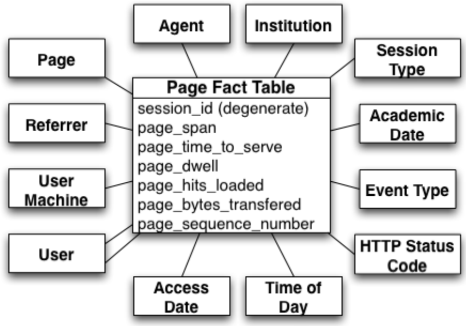


Fig. 1. Web Page Fact Table

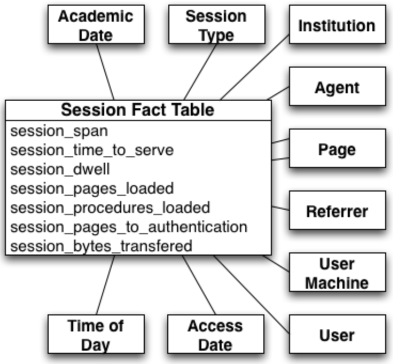


Fig. 2. Web Session Fact Table

dimension. This dimension has one hierarchy with three levels: Hour, Minute and Second. It has a record for each second of a day.

Academic Date. An academic calendar is usually associated with different structures that differ on the number of modules (semesters, four month periods and trimesters). Each of these structures is a different hierarchy, each with five levels: Year, Module (6, 4 or 3 months), Period (classes or examination period), Week (variable length, defined internally by each institution) and Day. It still has another hierarchy related to academic sessions, which are specific periods, of variable length, in an academic calendar. For example, the University Day (UPorto's anniversary day) is a one day academic session. All vacations are academic sessions. This hierarchy has three levels: Year, Session and Day. The Session level has information about the start and end date of the session, the session type (with classes, without classes, vacancies) and the number of days in the session.

User. This is a crucial dimension to the segmentation of users and to behaviour analysis. Accesses can be made by human users (identified or anonymous) or web crawlers. Identified users are students or workers (faculty or staff). Anonymous users are those who access the HEWIS without signing in. Comparatively, with WIS that gather information from online registration forms, HEWIS have the advantage of having more trustworthy information about identified users, as they usually obtain user's information in the student's school registration or in workers' act of contract. This dimension saves information about the user's academic degree, age group, gender, civil status, activity status, birthplace, role and department/service.

User Machine. The User Machine dimension gathers information about the physical geography (country) and web geography (top level domain, domain) of the machine that generates the web request. It also has information about the machine's location regarding the institution and the university and the access nature (for example: structured network, wireless network).

Agent. This dimension keeps information about the agent that has made the request, either a browser used by humans or a crawler.

Page. This dimension is an obvious one in WIS monitorization context. Although it has been modelled having SIGARRA in mind, it can be easily adapted to other types of HEWIS. It has one hierarchy with four levels: Application, Module, Procedure and Page. An application is an autonomous software artefact with one or more modules. Modules are logical units of the main functionalities and can be seen as a set of related procedures. A procedure generates pages and is the conceptual unit of interaction with the user. The same procedure generates different pages if the received arguments are distinct. For instance, the official pages of department A and department B are both generated by the same procedure.

Referrer. This dimension describes the page that has preceded the current access. This information is gathered from log files and is related to the domain of the referrer and the referrer itself: port, procedure (if it belongs to the HEWIS), query (everything that follows the '?' in an URL), the identification and description of the search engine (if this is the case) and the complete URL.

HTTP Status Code. This dimension has the category of the HTTP Status Code (Informational, Success, Redirection, Client Error, Server Error) and the description of the HTTP status code returned in the request.

Session Type. Here, web sessions are aggregated into predefined types of sessions. It has one hierarchy with several levels: session context (for example: enrolment in a course), local context (for example: consulting information of a course) and the final state of the session (if its main goal has been achieved).

Event Type. This dimension has just one hierarchy with one level and it describes what happened in a page at a specific time (for example: open a page, refresh a page, click a hyperlink, enter data in a form).

Institution. Information about academic institution associated with the web request is stored in this dimension.

2.3 Fact Tables

Each line in the Page Fact Table (see Figure 1) corresponds to a page served by the HEWIS. The session_id degenerate dimension is used to group pages in sessions. The double connection to the User dimension is explained by a SIGARRA's functionality that allows a user to act on behalf of another user (for example, course grades may be inserted by the faculty's secretary). The fact table has 6 measures: page time to serve (number of seconds taken by the web server to process all requests related to this page), page dwell (number of seconds the complete page is visible in user's browser), page hits loaded (number of resources loaded for the presentation of the page), page bytes transferred (sum of the bytes loaded in all the resources related to this page) and page sequence number (the sequence number of this page in the overall session).

A line in the Session Fact Table (see Figure 2) records the occurrence of a session in the HEWIS. A session is a set of page accesses, in a single browser session, by the same user, requested in intervals with less than 30 minutes. The double connection to the Page dimension allows the identification of the entry and exit pages of a session. The time related dimensions are associated to session's first request. The referrer dimension records the session's first referrer. This fact table measures are: session span (number of seconds between the first request and the complete load of the last request), session time to serve (number of seconds taken to serve all the requests in the session), session dwell (number of seconds of visibility of all the pages in the session), session pages loaded (number of pages in the session), session procedures loaded (number of distinct procedures in the session), session pages to authentication (number of pages until authentication; if there isn't any, this measure equals session pages loaded) and session bytes transferred (number of bytes transferred in this session).

3 SIGARRA Case Study

Although SIGARRA is defined at the institution level and is supported by several database and web servers, similarities between the HEWIS's structure in the several institutions and the nature of a data warehouse suggest the adoption of a centralized architecture at the university level for the data webhouse.

A prototype of a data webhouse has been built to monitor SIGARRA's usage in UPorto's Engineering Faculty, the institution where it is most used. As SIGARRA uses Oracle as its database management system (DBMS), this was also the underlying DBMS used in the staging area and in the data webhouse. They both co-exist in a single machine, independent of SIGARRA's machines.

A three month period of clickstream data has been loaded into a data webhouse with the dimensional model described before. As expected, after the webhouse load, the fact tables are the largest tables (Page fact table has 8 607 961 records and Session fact table has 984 848 records), followed by the Page (497 865 records), Referrer (461 832 records) and User Machine (202 898 records) dimensions. While log files from a 3 months period needed almost sixteen gigabytes of space (15,68 GB), the data webhouse with usage data from the same period needs almost three gigabytes (2,57 GB), a meaningful reduction of 83,6%.

3.1 Extraction, Transformation and Loading

The ETL involves getting the data from where it is created and putting it into the data warehouse, where it will be used. The architecture defined for the ETL process has three types of data sources: clickstreams, SIGARRA's database and other sources. The first come from web servers logs. SIGARRA's database is essential to gather information about the institutions, their internal organization (departments, sections, etc.), academic data (academic calendar, academic events, evaluation periods), HEWIS application structure, users and other kind of data (countries, councils, parishes, postal codes, etc.). The last data source includes data such as IP ranges of each type of access (wireless, structured network, etc.), data relating IP addresses with geographical areas, domain names, HTTP status codes, and information on search engines, browsers, crawlers, platforms and operating systems.

The extraction phases are the first to occur. At this phase, all data is extracted from its source and is transferred to the staging area with a simple file transfer.

Then, web servers' logs must be joined, parsed and transferred to the staging area. Parsing is done by a Perl script that has a web log file as input and generates a tab-delimited file with several fields and includes host IP address resolution, URL and referrer parsing, search engine, browser, crawler and operating system identification and cookies parsing.

After data loading into the staging area, clickstream is processed through PL/SQL, using a relational database. This process involves IP address/country resolution, session, page and user processing. Session and user tracking is based on session cookies, thus it is necessary to overcome the absence of cookies in first requests. A period of 30 minutes of inactivity will lead to a new session as proposed by several authors [5, 2, 13]. A change in the user associated with the session will lead to the same result. Users tracking must also deal with authentications that occur in the middle of a session.

Dimensions have been built with information from the tab-delimited file generated by the clickstream parsing and from SIGARRA. Fact tables have been built after dimensions due to the dependencies between them. The webhouse loading is done by copying the data from the staging area to the posting schema. At the end, all records that belong to a closed session are deleted from the staging area. A session is closed if it does not have requests in the last 30 minutes of a day (sessions going on near the end of the day may continue on the following day and must be processed by the next ETL iteration).

3.2 Data Analysis

The data analysis process has been made using Structured Query Language (SQL) and On-Line Analytic Processing (OLAP). Due to the star structure of the dimensional model, the queries were simple and had a good performance. Data analysis led to a detailed characterisation of SIGARRA's usage in several categories according to the main user types (students, faculty, staff, anonymous users, crawlers). Some of the results will be described next.

Time Related Analysis of Sessions and Pages. The average number of sessions by day is 10 942 and its distribution by user type is as presented in Figure 3. Excluding crawlers, the average session time span is 10,89 minutes, being the staff's sessions the longest ones (Figure 4). The average number of pages accessed by day is 95 575 and its distribution by user type is as presented in Figure 5. Excluding crawler's sessions, the average number of pages by session is 7,7.

Session Referrers. In the overall sessions, 79,23% were direct entries and 15,50% had origin in search engines, being Google the most used (99,8% of all search engines sessions).

User Machines. There were 155 distinct access countries. Staff and faculty users access mainly from inside the institution and anonymous users from outside (Figure 6). Inside institution, most accesses are from the structured network.

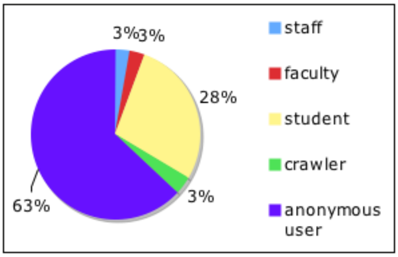


Fig. 3. Distribution of sessions by user type

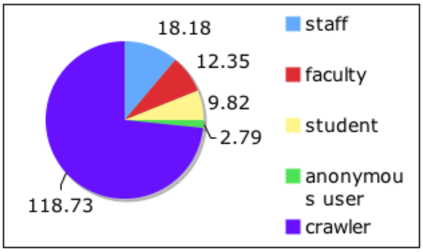


Fig. 4. Session span in minutes by user type

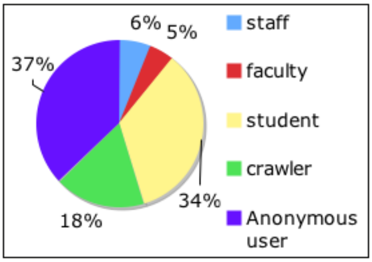


Fig. 5. Distribution of pages by user type

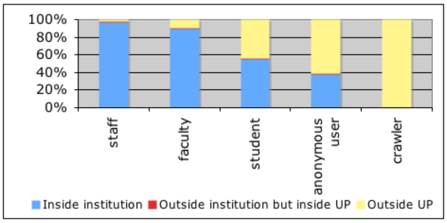


Fig. 6. Access type by user type

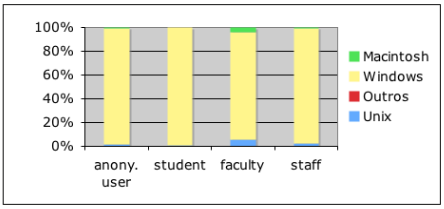


Fig. 7. Platforms used by user type

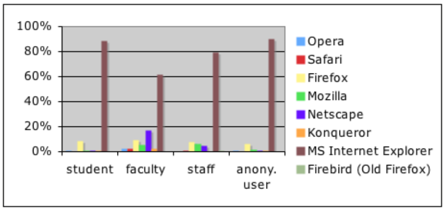


Fig. 8. Browsers used by user type

Access Agent. Windows is the most used platform. As it can be seen in Figure 7, faculty also use Unix and Macintosh platforms, although in a much smaller scale. As can be seen in Figure 8, the most used browsers are MSIE (88,10%) and Firefox (7,16%). Firefox use is growing and the inverse is happening with MSIE. The crawler with more sessions and pages requested is Googlebot (47,86% of all crawler's sessions and 86,89% of all crawler's requests for pages).

Number of Sessions by User Profile. There were 8 004 distinct users in the analysed period, 7 169 were students, 416 faculty and 252 staff users. The number of sessions is higher in users with less than 20 years, in undergraduate programme's students and particularly in the first curricular years of undergraduate programmes.

HEWIS Navigation. Student, programme and institution modules are the most used ones. Students and anonymous users have similar preferences in pages viewed. Two of the main entry pages are: Dynamic Mail Files (due to following hyperlinks to files in dynamic e-mail received) and Computer labs first page (because it is loaded in the background of every lab's computer). Authentication is mainly done in home page's authentication area (64,59% of all authentications) and the main underlying motivations are access to Dynamic Mail Files, Legislation, Summaries and Courses. The Help module is mainly used by anonymous users.

Specific Pages Usage. Home page connections most used are: authentication, search and programmes. The two undergraduate programmes most viewed by anonymous users are: Electrical and Computers Engineering and Informatics and Computing Engineering and the two master programmes most viewed by anonymous users are: MsC in Informatics Engineering and MsC in Information Management. The main searchs are related to students, staff and courses.

3.3 Recommendations

The analysis has allowed the detection of some unusual access patterns: too long anonymous user's sessions (about 170 000 pages requested over 3 days) with a name of an institution's machine; abnormally large processing time in a specific day of the period analysed. It has also allowed the production of some improvement recommendations. It should be created a direct connection to study plans of each programme in the programmes lists (due to the frequent path: programme list / programme page / study plan / course page, that suggests the course page is distant from the home page). The initial page of the student's module should be used to provide information and communicate with students (this is the 6th page most viewed, specially by anonymous users and students). Help module usage by faculty should be stimulated (these users rarely uses this module, preferring the phone). The complexity and usability of the pages from where an higher access to the help occurs should be analysed. Marketing strategies to promote the programmes with less page views should be developed. In order to minimise changes and 404 type error code (Not Found), it should be used URL independent from the underlying technology (79,12% of 404 error code are direct entries in the HEWIS which suggests the use of bookmarks with broken links due to URL changes). The procedures where the 505 (Internal Server Error) error code has most occurred should be analysed.

4 Conclusions and Future Work

Data webhouse systems are here presented as a solution to monitor the use of Higher Education Web Information Systems (HEWIS). This paper pretends to enlarge the application and study of webhousing systems to the academic context. Despite the similarities between Web Information Systems, there are differences between HEWIS and e-commerce sites, being the last frequently used to exemplifications and instantiations of data webhouses. While HEWIS pretends to archive and register higher education activity and has features adapted to this scope, in e-commerce sites the main intent is to sell products and has a well defined set of procedures is available (add to shopping cart, insert payment information, etc.). As they have different goals and scopes, the relevant information is different (for example: the Academic Date dimension doesn't make sense in an e-commerce site webhouse and its very important in an HEWIS webhouse), what justifies a different dimensional model.

It was described a dimensional model to monitor HEWIS's usage. This model has been implemented in a data webhouse prototype to monitor UPorto's HEWIS usage. In the development of this prototype it has been defined an extraction, transformation and loading architecture that, with adaptations to specific data sources, can be used in similar contexts.

The prototype developed proved the usefulness of data webhouses to WIS and more specifically to HEWIS. It allowed the generation of knowledge on SIGARRA's user behaviour, the detection of abnormal situations and the definition of a set of recommendations. It was also possible to verify that there is a significant reduction in the amount of disk space required to store web usage data, what stimulates the storage of web usage data in a dimensional model. On the other hand, it has demonstrated the analytic flexibility of data webhouses, an advantage when compared to other monitorization techniques. Also, it showed that queries executed on a star dimensional model with a meaningful amount of data have a good performance.

Future work involves applying data mining techniques that allows user clustering based on navigation paths or preferences, navigation patterns discovery, detection of set of pages that have more probability of being together in the same session and user classification based on predefined parameters.

References

1. Jesper Andersen, Anders Giversen, Allan H. Jensen, Rune S. Larsen, Torben Bach Pedersen, and Janne Skyt. Analyzing Clickstreams Using Subsessions. In *Proceedings of the 3rd ACM International Workshop on Data Warehousing and OLAP*, pages 25–32. ACM Press, 2000.
2. Bettina Berendt and Myra Spiliopoulou. Analysis of Navigation Behaviour in Web Sites Integrating Multiple Information Systems. *The VLDB Journal*, 9(1):56–75, 2000.
3. M. S. Chen, J. S. Park, and P. S. Yu. Data Mining for Path Traversal Patterns in a Web Environment. In *Proceedings of the 16th International Conference on Distributed Computing Systems (ICDCS ́96)*, page 385. IEEE Computer Society, 1996.
4. Robert Cooley. The Use of Web Structure and Content to Identify Subjectively Interesting Web Usage Patterns. *ACM Trans. Inter. Tech.*, 3(2):93–116, 2003.
5. Robert Cooley, Bamshad Mobasher, and Jaideep Srivastava. Data Preparation for Mining World Wide Web Browsing Patterns. *Knowledge and Information Systems*, 1(2), 1999.

6. Magdalini Eirinaki and Michalis Vazirgiannis. Web Mining for Web Personalization. *ACM Trans. Inter. Tech.*, 3(1):1–27, 2003.
7. Karuna P. Joshi, Anupam Joshi, Yelena Yesha, and Raghu Krishnapuram. Warehousing and Mining Web Logs. In *Proceedings of the Second International Workshop on Web Information and Data Management*, pages 63–68. ACM Press, 1999.
8. Ralph Kimball and Richard Merz. *The Data Webhouse Toolkit*. John Wiley & Sons, Inc., 2000.
9. Ralph Kimball, Laura Reeves, Margy Ross, and Warren Thornthwaite. *The Data Warehouse Lifecycle Toolkit*. John Wiley & Sons, Inc., 1998.
10. Ron Kohavi. Mining e-Commerce Data: The Good, The Bad, and The Ugly. In *Proceedings of the seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 8–13. ACM Press, 2001.
11. Richard Li and Jon Salz. Clickstream Data Warehousing. *ArsDigita Systems Journal*, 2000. Available from: `http://www.eveandersson.com/arsdigita/asj/clickstream/` [cited 2005-09-11].
12. Brij M. Masand, Myra Spiliopoulou, Jaideep Srivastava, and Osmar R. Zaiane. WEBKDD 2002: Web Mining for Usage Patterns & Profiles. *SIGKDD Explor. Newsl.*, 4(2):125–127, 2002.
13. Jaideep Srivastava, Robert Cooley, Mukund Deshpande, and Pang-Ning Tan. Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data. *SIGKDD Explor. Newsl.*, 1(2):12–23, 2000.
14. Mark Sweiger, Mark R. Madsen, Jimmy Langston, and Howard Lombard. *Clickstream Data Warehousing*. John Wiley & Sons, Inc., 2002.
15. Tak Woon Yan, Matthew Jacobsen, Hector Garcia-Molina, and Umeshwar Dayal. From User Access Patterns to Dynamic Hypertext Linking. *Computer Networks ISDN System*, 28(7-11):1007–1014, 1996.

A User Management System for Federated Databases Using Web Services

Fuyu Liu[1] and Erdogan Dogdu[2]

[1] School of Computer Science, University of Central Florida, Orlando, Florida, USA
fliu@cs.ucf.edu
[2] Computer Engineering Dept., TOBB Economics and Technology University, Ankara, Turkey
edogdu@etu.edu.tr

Abstract. A federated database system (FDBS) is a distributed system that consists of a number of autonomous and heterogeneous database management systems (DBMS). Administration of a FDBS is a challenging task due to the heterogeneity of database management systems in the FDBS, heterogeneous platforms these DBMS are deployed on, and non-standard access protocols these systems provide. One of the important tasks in the management of FDBS is user management. In this paper we propose a new architecture for user management in FDBS, based on the "web services" middleware. The system includes a Central Controller for maintaining a directory of component databases, database access roles, and users. Each database component and the Central Controller are accessed via web services providers that are deployed on each component site. These web services are lightweight interfaces hiding the heterogeneity of different platforms. The system is highly scalable and portable. New DBMS can be easily added to the FDBS after the web services interfaces for the regarding DBMS are installed on the component sites.

1 Introduction

Information integration and process automation are two top-priority challenges in the business world. These challenges are mostly met via customized programming which is expensive, difficult, and error-prone. In this paper, we address both issues in the context of federated database systems.

Information integration problem is solved with two approaches. Either (1) all databases are consolidated into a central location, processed, and made accessible to the interested parties, or (2) data is accessed "in place" via technological solutions such as a federation technology. First solution, consolidation, is an expensive one and it also does not provide real-time data. With the recent developments in networks, technology, and distributed computing, now it is easier to realize the second approach. In this paper, we attempt to extend the federation technology via recent technological developments in distributed computing middleware. In our study, "web services" middleware is used as the enabling technology to advance the user management and access control in federated database systems.

A Federated Database System (FDBS) is composed of a number of heterogeneous databases and is usually distributed [1, 3]. The differences among these component

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 88 – 97, 2006.

databases could originate from different types of databases or from different versions of the same database system type. The heterogeneity in federated databases poses challenges in building a federated database management system (FDBMS) [4]. An FDBMS needs to support applications or users submitting SQL statements referencing more than one database in a single statement. Access control becomes quite important here. While applications or users try to access data from databases, FDBMS should check if the application or the user has the appropriate access privileges to specific DBMSs or not. User maintenance in a federated database system is also a challenging task. Typically, for different types of DBMSs, different client tools are needed to administer databases remotely. But, in a federated database system this becomes quite cumbersome; there is not a single integrated view over the whole system.

While still in its early stage, web services [2] technology and the resulting Service Oriented Architecture (SOA) is becoming a popular choice in building distributed software systems. A web service is a remote application that is accessible via standard Internet protocols. Web services middleware is not a completely new idea, but it demonstrates a new way of using software over the Internet and has a promising future.

In this paper, we propose a new infrastructure to solve the user management and access control problem in federated database systems using web services technology. In our design, each database server is equipped with a unique and secure add-on web services interface for user and access control. Once each member database system is enabled with this standard user and access control mechanism, managing databases can be done remotely from a designated central control mechanism, or from any other remote application. Due to the unique advantages of web services technology, our proposal is independent from any specific database management system technology and its propriety interfaces, therefore providing a standard and easy to use mechanism for user management and access control in federated database systems.

The rest of the paper is organized as follows: in Section 2, we present the background on federated database systems and web services. Related work follows in Section 3. In Section 4, we discuss our proposed solution in detail. Finally, Section 5 concludes the paper.

2 Background

In this section, we talk about access control in federated database systems first, and then give a brief background on web services.

2.1 Access Control in Federated Database Systems (FDBS)

A federated database system integrates existing and possibly heterogeneous databases while preserving their autonomy. Access control refers to the access rights and their control in a DBMS. Access rights are those such as read, write, update, delete operations on database artifacts such as tables and views, and the control of these privileges involves operations like grant and revoke for users and user groups. Access control is a difficult issue in a federated database system. Users may have different privileges in accessing data in different databases; therefore, there should be some kind of access control mechanism to deal with the security problems in these distributed environments.

In a loosely coupled federated system, security problems are similar to those in traditional databases; each component database handles its own access control [5]. Because there is no federation authority, a security policy for the federation does not exist. In a tightly coupled federated database system, on the other hand, a federation authority exists and it has its own access control mechanisms. Access to data can be seen at two different levels: at the federation level, where users explicitly require access to the federated data; and at the local level, where local requests corresponding to global requests must be processed. Access control can be executed at both levels [3].

2.2 Web Services

Web services can be described as any functionality that is accessible over the Internet using XML messages in the communication protocol. The most important underlying architecture of web services is Service Oriented Architecture (SOA). An SOA focuses on how components are described, integrated and organized together to support the automatic and dynamic discovery, binding, and usage of web service functionalities. There are three major roles in a typical SOA architecture: a service provider, a service broker, and a service requestor.

Currently, web services framework consist of at least the following protocols: SOAP, WSDL, and UDDI. SOAP (Simple Object Access Protocol) is a lightweight protocol based on XML for exchange of information in a decentralized, distributed environment. WSDL (Web Service Description Language) is also XML based. The purpose of WSDL is to describe web services in a standard way. After a web service is published, a Universal Description, Discovery and Integration (UDDI) registry serves as a public repository for web service information.

3 Related Work

Mehrotra and coauthors first proposed the idea of providing database as a service [14]. In that paper, they focused on how to provide services to access one single database. More recently, Thakar et al proposed SkyQuery [15], which utilizes web services to answer queries in federated databases. They suggest a good algorithm to evaluate a probabilistic federated spatial join query. Zhub et al try to exploit web services to support dynamic data integration in a federated environment [17].

Access control in distributed systems is also a popular research topic. Bertino et al propose an XML-based access control language (X-RBAC) which provides a framework for specifying mediation policies in a multi-domain system [16]. Bertino et al later extend the X-RBAC language to support temporal role based access control [19]. In [18] Barker et al exploit the usage of the formally specified RBAC policies to support federated relational database access over the network. For grid computing environments, Raman et al present a layer of services providing data transparency to end users and enable ease of information access [12]. Other than the access control problem, in [11] Chun et al discuss the trust management problem in a federated system and propose a layered architecture to address the problem.

There are also a number of commercial tools available as Federated Database Management Systems [6, 7, 8, 9, 10]. Unfortunately, these systems either do not address the user management problem or do not support access control policies across the whole federated system.

4 Federated Database User Management System

In the previous two sections, we covered the background and related work. Although there are some commercial products available as Federated Database Management Systems, they are either too expensive or do not provide a generic way to access databases remotely.

Considering the benefits web services provide, such as the ability of invoking methods remotely via standard web protocols (like HTTP), in this paper, we propose a Federated Database User Management System using web services technologies. In this system a web service is deployed on each one of the component database system. Administrators or ordinary database users can access component database systems via standard web service calls. Only a simple web service client is needed to consume these web services deployed. There is no need for other remote access clients. Also, in our system services like granting global privileges to users at other databases are enabled. By combining access control abilities provided by individual databases and global privileges enforced by our system, we can realize a federated access control mechanism for the whole federated database system.

In this section, we will first introduce the features of our system. After that, the architecture of our system will be presented. Then, the data that needs to be saved in the system will be discussed in detail. The next two subsections go over the deployed web services and the client program. How to deploy the system is discussed in the end of this section.

4.1 Features

The following are the main features of our Federated Database User Management System:

a. Manage database users remotely as a database administrator. Most commands issued by DBAs are supported.
b. Grant privileges remotely as an ordinary user. Privileges include local privileges granted to users residing on the same database and global privileges granted to users on other databases.
c. Add new databases or delete existing databases from the Federated Database System.
d. Access control support for global query execution.

4.2 Architecture

The whole system is composed of three modules: Client, Central Controller, and Component Database (Fig. 1).

In Figure 1, Web Service-I (WS-I) is the Central Controller Web Service. This web service is responsible for adding databases to the system, removing databases from the system, and storing access control information. Web Service-II (WS-II) is the Remote DBA Service. WS-II communicates with the component database directly via Java Database Connectivity API (JDBC), or similar technologies such as Open Database Connectivity (ODBC).

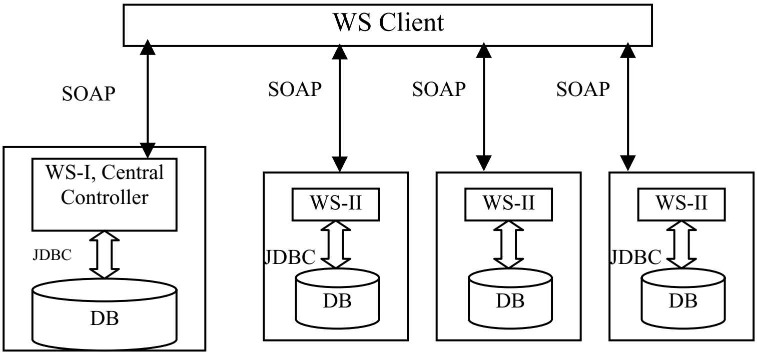


Fig. 1. Architecture of User Management System

With WS-II, database administrator could manage users and roles on individual databases and ordinary database users could grant privileges to other users located on the same database (or revoke them). After reading database information from Central Controller, the web service client talks to individual databases directly. In this framework, the Central Controller has a role similar to a directory service in SOA.

4.3 Data Stored in Central Controller

In order to keep database and user access control information, we need to have a database at the Central Controller to store that information. This information could be accessed and modified by the web services deployed on the Central Controller (WS-I). Three tables will be used to keep the information we need: *dbs*, *ccusers*, and *ccprivileges*. *dbs* table is used to store information about component databases. *ccusers* table holds information about those database users who have granted global privileges to users from other databases. *ccprivileges* table is used to store user access privileges information. Schemas for the three tables are given below:

```
create table dbs (
  logicalname  varchar2(10) primary key,
  type       varchar2(20),// database type
  location   varchar2(30),// physical address
  name       varchar2(20),// database name
  portNumber   integer);
create table ccusers (
  logicalname  varchar2(10) references dbs,
  username  varchar2(20),
  password  varchar2(20));
create table ccprivileges (
  grantordb varchar2(10) references dbs(logicalname),
  grantor   varchar2(20),
  privilege varchar2(20),
  objectname   varchar2(50),
  granteedb varchar2(10) references dbs(logicalname),
  grantee   varchar2(20));
```

Both username and password are stored in the *ccusers* table such that whenever a user issues a global query to access another user's object, the second user's password could be utilized to get the requested object. Passwords are encrypted to enhance security. The *ccprivileges* table is to store all global privileges granted. Each time when a new global privilege is granted, grantor's information in the *ccusers* table will be created if it does not exist or be updated if it exists. Each time a global privilege is revoked, the system will check the *ccprivileges* table to see if there are access privileges granted by that grantor. If there is no other access privileges granted by the same grantor, the grantor's entry in the *ccusers* table will be removed for security reasons.

In order to access the three tables stored in the database, we need to know how to connect to the database and more importantly, the username and password to access that database. We use an XML file for this purpose. Whenever it is needed, this information is retrieved by the web services deployed on the Central Controller and used afterwards.

4.4 Deployed Web Services

We present the two deployed web services in detail in this section. The Remote DBA Service is deployed on the site of each component database. Different types of database platforms will have different types of implementations for this service. The Central Controller Service is used for manipulating databases, and granting/revoking global privileges.

4.4.1 Remote DBA Service

Available methods in this web service are: *createUser, deleteUser, modifyUserPasswd, viewAllUsers, createRole, dropRole, viewAllRoles, grantRole, revokeRole, grantPrivilege, revoke-Privilege, authenticateUser.*

Considering the similar syntax used while granting roles and granting system privileges (e.g. connect, resource), we use methods *grantRole* and *revokeRole* to take care of system privilege manipulations. Methods *grantPrivilege* and *revokePrivilege* will be used to manipulate object privileges (e.g. select, update, delete).

To demonstrate parameters used in these methods, we give one example here. The following example shows the parameters used in method *createUser*. There are seven input variables:

```
createUser (
  String dbLocation, String dbName, String portNumber,
  String username, String passwd,
  String newUsername, String newUserPasswd)
```

The first five parameters represent database location, database name, port number, user name, and user password respectively. These parameters are required for most methods in this web service. New user name as well as a password is needed for creating a new user. All other methods have similar input variables.

4.4.2 Central Controller Service

The following methods will be deployed in this service.

a) *grantPrivilege*: This method is used to grant global privilege. For instance, user1 at db1 wants to grant certain privilege (e.g. select) on his/her table1 to user2 at db2. There are seven input parameters for this method as shown below:

```
grantPrivilege (
  String grantorDBLocation, String grantor,
  String grantorPasswd, String privilege,
  String objectName, String granteeDB, String grantee)
```

Parameters grantorDB, grantor, privilege, objectName, granteeDB, and grantee will be stored in the table *ccprivileges* at the Central Controller. Parameters grantorDB, grantor, and grantorPasswd will be stored in the table *ccusers* at the Central Controller.

b) *revokePrivilege*: The opposite of method *grantPrivilege*

c) *viewAllPrivileges*: To view all global privileges in the whole FDB system.

d) *checkPrivilege*: To check if the requested privilege exists or not.

e) *viewPrivilegeByGrantor*: Used by the Central Controller's administrator to view privileges granted by certain grantor. Grantor is identified by the username and the name of database where the user is located.

f) *viewPrivilegeByGrantee*: Used by the Central Controller's administrator to view privileges received by certain grantee.

g) *viewPrivilegeGrantedByMe*: Used by grantors to check privileges that they grant to other users.

h) *authenticateUser*: To authenticate Central Controller's administrator. Only the Central Controller's administrator has the privilege to add a new database or delete a database.

i) *viewAllDBs*: To pull out all available databases in the whole FDB system.

j) *addDB*: To add a new database to the whole FDB system. Only the Central Controller's administrator has the privilege to invoke this method.

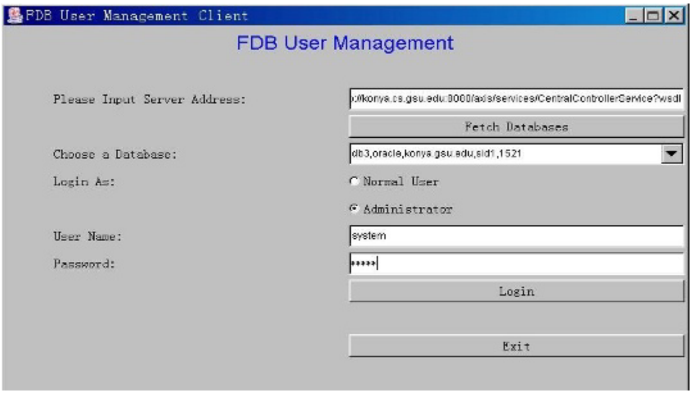
k) *deleteDB*: The opposite of method *addDB*.

4.5 Client Program

Client program serves as a prototype tool to consume the two web services introduced above. The interface of the client program consists of a series of four screens. The first screen is called Login Screen, where a user or an administrator can access all databases in the FDB system, and then login into a selected database. The second screen is Administrator Screen, designed for the database administrator. The third screen is User Screen, which is for ordinary database user. The fourth screen is Central Controller Screen, which is used only by the Central Controller's administrator to manage the whole system.

4.5.1 Login Screen

After the client program is launched, user will be presented with the Login Screen (Figure 2). In this screen, user is asked to provide the address of the Central Controller web service. Then the user can select one database from the list and then login into the selected database either as a normal user or an administrator. From the database list, user can also choose the Central Controller server. In this case, user must login as an administrator to manipulate existing databases and view existing global privileges.



FDB User Management Client

FDB User Management

Please Input Server Address: p://konya.cs.gsu.edu:8000/axis/services/CentralControllerService?wsdl

Choose a Database: db3,oracle,konya.gsu.edu,sid1,1521

Login As: ○ Normal User ● Administrator

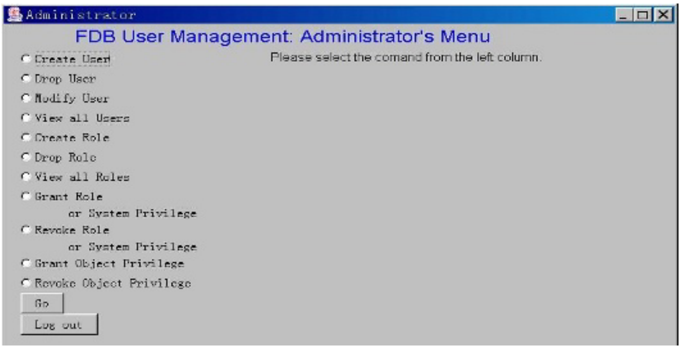
User Name: system

Password: *****

Fig. 2. Login Screen

4.5.2 Administrator Screen

An administrator screen is provided for database administrators as shown in Figure 3. A series of commands are available to the administrator on the left column. Administrator can manage users, roles, and privileges. Administrator needs to select a command from one of these radio buttons on the left column. To make the user management task easier, our system also enables administrators to view all users/roles in a database.



Administrator

FDB User Management: Administrator's Menu

Please select the comand from the left column.

Fig. 3. Administrator Screen

4.5.3 User Screen

If a user chooses to login into a database as an ordinary user, the User Screen is displayed. There are five options available to an ordinary user. User can grant/revoke local or global privileges and view all global privileges where the user acts as the grantor. Granting global privileges is to give users from other databases the permission to access data on this database, which is quite important for information sharing and access control in Federated Database Systems.

4.5.4 Central Controller Frame

While in the Login Frame, user could also choose the Central Controller to login in. In this case, user is required to login as an administrator since ordinary users do not have privileges to manage data in Central Controller. Central Controller's administrator can view all databases in the whole FDBS. Administrator can add databases to the system or delete databases from the system. To help the management of global privileges, we provide three options here: view all global privileges in the system, view global privileges based on grantor's name and view privileges based on grantee's name.

4.6 Implementation and Deployment

We implemented this Federated Database User Management System using Java technology. Two different versions of client programs were developed. One is a stand-alone Java-based program, and the other one is a web-based system. Apache Axis [13] is used for the development, test, and deployment of the web services.

To deploy web services in an Apache Axis environment, we need to write a Web Service Deployment Descriptor (WSDD) file first to specify the names for the web services, and use that file to deploy web services. Undeploying web services is just the opposite. An undeployment descriptor is needed to indicate the service names to be removed.

5 Conclusions

In this paper, we designed and implemented a Federated Database User Management System using Web Services technology. In this Federated Database system, a server is used as a Central Controller, where all information about individual databases and global privileges are stored.

Two kinds of web services are developed to implement this system. The Remote DBA Service is for a database administrator or an ordinary user to access databases remotely via standard web services calls. This service is deployed on the server of each database component on top of Apache Axis platform. The Central Controller Service is for administrating the Central Controller of FDBS, accessing and modifying data stored on the Central Controller.

To the best of our knowledge, Federated Database User Management System is the first implementation of its kind using Web Services technology. Web Services technology provides flexibility and interoperability to this system. Deployed web services can be integrated with other web services. Extensions can be easily made based on the original web services. The system is scalable and portable. To add an extra database to the federated database system, one only needs to add the new database information to the Central Controller and deploy the corresponding RemoteDBAService on the newly added database machine.

References

1. Amit Sheth, James Larson, "Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases", *ACM Computing Surveys*, 22(3), pp 183-236, 1990.
2. "*Web Services Activity*", http://www.w3.org/2002/ws
3. S.D.C di Vimercati, P. Samarati, "Access Control in Federated System", *ACM New Security Paradigm Workshop*, Lake Arrowhead, CA, 1996.
4. Mario Piattini, Oscar Diaz, *Advanced Database Technology and Design*, 2000 Artech House Inc.
5. Dirk Jonscher , Klaus R. Dittrich, "An Approach For Building Secure Database Federations", *Proc. of the 20th VLDB Conference*, Santiago, Chile, 1994
6. Marjorie Templeton, Herbert Henley, Edward Maros, Darrel J. Van Buer, "InterViso: Dealing With the Complexity of Federated Database Access", *VLDB Journal* 4(2): 287-317(1995)
7. Laura Haas, Eileen Lin, "IBM Federated Database Technology", *DB2 Developer Domain*, March 2002, see: http://www-106.ibm.com/developerworks/db2/library/techarticle/0203haas/0203haas.html
8. "*Heterogeneous Data Access*", www.oracle.com/gateways/
9. "Database Administration Suite for Distributed RDBMS: Heterogeneous Database Administration", http://www3.ca.com/Solutions/Product-Family.asp?ID=2858
10. Jayavel Shanmugasundaram, Jerry Kiernan Eugene, Shekita Catalina, Fan John Funderburk, "Querying XML Views of Relational Data", *Proc. of the 27th VLDB Conference*, 2001.
11. Brent N. Chun and Andy Bavier, "Decentralized Trust Management and Accountability in Federated Systems", *37th Hawaii Int. Conf. On System Sciences*, 2004
12. Vijayshankar Raman,Inderpal Narang, Chris Crone, Laura Haas, Susan Malaika, Tina Mukai, Dan Wolfson, Chaitan Baru, "Data Access and Management Services on Grid", *The Fifth Global Grid Forum*, 2002
13. "*Apache Web Services*", see: http://ws.apache.org/axis/
14. Hakan Hacigumus, Bala Iyer, and Sharad Mehrotra, "Providing Database as a Service", *Proceeding of ICDE*, 2002.
15. Tanu Malik, Alex Szalay, Tamas Budavari, and Ani R. Thakar, "SkyQuery: A Web Service Approach to Federated Databases*", Proceeding of CIDR Conference*, 2003.
16. James B.D. Joshi, Rafae Bhatti, Elisa Bertino, and Arif Ghafoor, "Access-Control Language for Multidomain Environments", *IEEE Internet Computing*, 2004
17. Fujun Zhub, Mark Turnera, Ioannis Kotsiopoulosc, Keith Bennettb, Michelle Russelld, David Budgena, Pearl Breretona, John Keanec, Paul Layzellc and Michael Rigby, "Dynamic Data Integration using Web Services", *IEEE ICWS* 2004
18. S. Barker, P. Douglas, "Protecting federated databases using a practical implementation of a formal RBAC policy", *Information Technology: Coding and Computing*, 2004
19. Rafae Bhatti, Arif Ghafoor, and Elisa Bertino, "X-GTRBAC: An XML-Based Policy Specification Framework and Architecture for Enterprise-Wide Access Control*", ACM Tran. On Information and Sys. Security*, 2005

A Dynamic Evaluation Framework for Mobile Applications

Anders Magnus Andersen and Torab Torabi

Department of Computer Science and Computer Engineering,
La Trobe University,
Bundoora, VIC, 3086, Australia
am5andersen@students.latrobe.edu.au, t.torabi@latrobe.edu.au

Abstract. Due to the large variation in capabilities of mobile devices and the lack of true standards, it is hard to develop applications for the mobile environment that will behave similar on all devices and in different environments. This article introduces the concept of a Dynamic Evaluation Framework that uses several different implementations for one specific application. The performance of each implementation is evaluated at runtime ensuring that the optimal implementation is always used. We describe the architecture and discuss the feasibility of the framework. As part of the evaluation we have developed a simple chat application with a seamless connection that evaluates and selects the optimal connection in real time. The evaluation technique is based on Goal-Question-Metric. The test environment is a J2ME CLDC application that transfers data with Bluetooth and GPRS over the JXTA network.

1 Introduction

Due to the rapid development of new technologies and the large diversity in quality of services provided to mobile users, it is increasingly complex to develop applications that will target all devices and environments. It is common to develop and maintain several instances of one application for different environments. Mobile devices are often used in such an environment that a single implementation can not work in every situation. It is for example common to say that you develop a *Bluetooth* application; the optimal would be to develop a *P2P* application, disregarding the method of connection between the peers.

We propose a framework that supports using several implementations for one specific application. The framework will evaluate the optimal implementation at runtime. The goal is to have the application optimised, based on few parameters passed to an evaluation component which will choose the optimal decision based on empirical values collected. It is basically a “try and fail” approach where the framework learns about the environment rather than the user configuring all the parameters for the application.

Imagine the scenario where a mobile user has a 3G subscription with fixed cost, and another user has a prepaid GPRS account with a very expensive usage cost. Even though the environment and device is similar, the application should be aware of this difference according to the environment. The application could download content for

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 98 – 107, 2006.

the GPRS-user when the user is close to a WLAN or a Bluetooth connection, and stay offline when these connections is not available, but for a 3G-user the application could download content in real time.

To evaluate the framework we have chosen to implement the Seamless Connectivity component, as mobile connectivity is subject to a lot of change and often is unstable. Mobile connections are often lost, especially with short range connections as Bluetooth [1], Infrared [2] and Wireless LAN [3]. The challenge is to use the optimal connection in all time, and provide a horizontal handoff technique [4] so that the transfer continues on alternative connection if the preferred connection is lost or becomes too slow. In this case the user does not have to worry about which connection to use; he will be only concerned about the delivery of his content. The cost factor is introduced as a variable for choosing the optimal connection as described in section 2.

The transfer protocol used in the evaluation is JXTA [5]. In JXTA peers can communicate regardless of platform, connection technology, firewalls and NAT's. Peers can create their own groups in the JXTA network and peers are not dependent on a central server. The prototype is developed for J2ME MIDP 2.0 devices using JXME [6] communicating via Bluetooth and HTTP.

Section 2 describes the architecture of the framework. In Section 3 Seamless Connectivity case study is used for evaluation of our proposed framework. Seamless Connectivity and the techniques used for switching between connections and choosing the optimal connection are described. Design and implementation is discussed in Section 4, and an evaluation and conclusion is given in Section 5 and 6 respectively.

2 Architecture

The proposed framework will translate information from different sources and present it in a uniform way, abstracting the implementation issues away from application logic. Few mobile applications incorporate this kind of behaviour. Most applications have a single implementation and do not choose an optimal solution when the environment parameters changes. By implementing one function in several different ways, one for each environment scenario, so the applications can achieve an optimal solution in different environments. It is necessary to evaluate which implementation is best for each case. An example of a dynamic evaluation framework function is changing from one transfer protocol to another due to varying transfer rates in different scenarios. Another function could be changing the connectivity due to breakdown or overload on the network. This behaviour would provide an optimal solution to the user regardless of changing conditions; the application becomes self-optimizing.

A framework has been proposed in [7], as a dynamic lightweight platform but does not provide any real time evaluation.

2.1 Components

The goal of the framework is to develop applications with a holistic behavior where the users don't have to specify the communication parameters, only the message. So if the user wants to send a message to a remote peer, he only specifies the message and the recipient, not the method of delivery. To achieve such a holistic behavior, the component needs to be fully in control of different implementations. We have identified four logical components with different responsibilities:

- **Implementation:** The implementation component is the component that performs the actual task. If could be an implementation of a transfer interface like Bluetooth, a data access object, an algorithm or any other form of functionality that has a clearly defined interface. It may require some custom programming to enable the Observer component to collect the necessary values for the optimal evaluation.
- **Observer:** An observer component is attached to each implementation and is responsible for recording data about the implementation component as it performs tasks. This can be implemented in to ways, either the observer is placed as a facade for the implementation and records the data that goes through it or the observer can be a passive observer and the implementation pushes data to the observer that stores it for easy retrieval for the Controller component.
- **Evaluation:** The Evaluation component simply executes the evaluation algorithm when called upon. If necessary, it keeps a record of historical values for use during the evaluation.
- **Controller:** The controller is the decision maker. It collects data from the observers, and performs and uses the evaluation component to choose the optimal

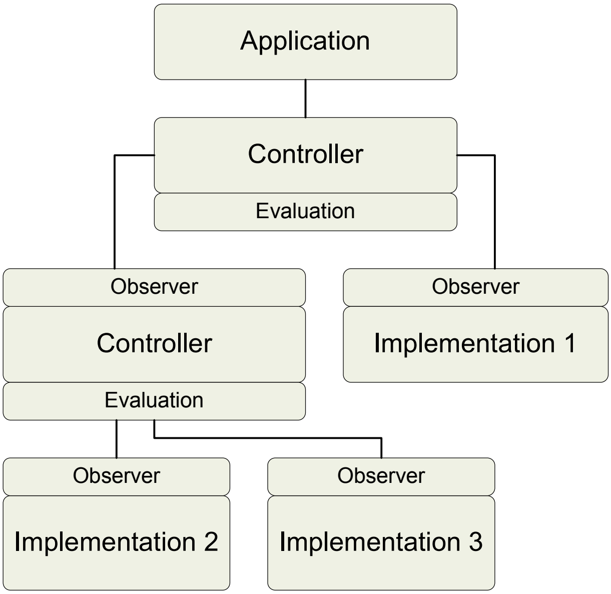


Fig. 1. A hierarchical evaluation structure for choosing the optimal implementation for a specific scenario. This figure shows 3 implementations capable of performing the same task but using different implementations. Implementation 2 and 3 uses the same technology but with slightly different implementations that performs different in different scenarios and are therefore treated as one component for the main controller.

implementation for every scenario. It is also responsible for starting and stopping services at the implementation and acts as an adaptor for differences in the implementations, giving the application and the user a uniform communication flow regardless of which implementation is being used.

Figure 1 shows an application with 2 holistic components, each with 2 implementations. One of the implementations is a holistic component that again contains two implementations. This might be two slightly different implementations of the same technology that has almost the same properties, but might use different algorithms or protocols that give a different result for different scenarios. One holistic component has no restriction on number of implementations it can maintain, but only two is used in this evaluation. The observer calculates the holistic value for each implementation and the controller chooses the optimal implementation based on these values.

These are examples of components that the proposed framework could contain.

- **Seamless Connectivity.** A connection manager that provides handling of multiple connections, vertical handoff and methods for choosing an optimal connection based on parameters such as connection speed and cost. See section 3.
- **Protocol translation.** Protocol abstraction enables different applications to communicate with each other. It translates messages from one message to another. (Example, a P2P client that transfers using several different P2P protocols).
- **Holistic Storage.** Save files on device memory, memory card, or on a remote disk. Data is stored on the best medium available.
- **Algorithm interface.** Use the observer to evaluate algorithms that behave differently in different scenarios.

We have implemented Seamless Connectivity as a case study to evaluate the feasibility of the Dynamic Evaluation Framework.

3 Seamless Connectivity

Imagine that you are transferring a file from your mobile phone to your desktop computer. The transfer starts with Bluetooth because you have configured the application to use the cheapest connection available. But when you move out of range from your home computer, you would like that the transfer will not fail; instead it will complete the transfer using GPRS or 3G.

The Seamless Connectivity component takes care of connection specific tasks, like Bluetooth pairing and managing several connections.

For a connection manager to achieve a truly holistic behaviour it should have control over connectivity and be able to choose which implementation to use. It also should be able to adapt to environment changes, always using the optimal connection.

Such framework have been proposed by the authors of [8] and [7], but they have not provided an evaluation using J2ME MIDP 2.0.

3.1 Choosing the Optimal Connection

Previous research have proposed an evaluation algorithm for the mobile environment [9]. We propose a high level, simpler empirical algorithm with real time evaluation and with minimum configuration and also with an architecture supporting a hierarchy of controllers.

Criteria for choosing the optimal connection has been discussed in [10] [11] [9]. We choose to narrow down the set of criteria to include data transfer rates, connection initiation/handshake delay, availability, cost, and user preference. Whether or not a connection is already connected is also considered. Battery consumption is not included as a parameter at this point will not probably affect the final result. We present formula (1) for choosing an optimal connection.

$$V_c = ((T_i C_o) + (\frac{M_{kb}}{KB/s}))(1 + C_b K_c)a \quad (1)$$

V_c is the optimal connection value. The connection with the lowest value is the preferred connection but the connection should only be considered if $V_c > 0$.

$(T_i C_o)$ is the initiation time where T_i is the time to initiate in seconds, and $C_o = 0$ if there is a connection and $C_o = 1$ otherwise. C_o cancels out the initiation time if the connection already is connected. This is described in more detail in section 3.3. $(\frac{M_{kb}}{KB/s})$ calculates how long transfer time that is left where M_{kb} is how much of the message that is left to transfer and KB/s is the transfer rates in kilobytes per second. $(1 + C_b K_c)$ calculates the importance of cost. C_b is the price per kilobyte for this specific connection and K_c is the cost factor or the importance of the cost. The cost can be in any currency but the cost factor should be adjusted accordingly. A one is added to ensure that a connection with no cost doesn't zero out the equation. The last variable in the equation is the availability factor. $a = 0$ if connection is unavailable and $a = 1$ if available. If a connection is unavailable it will not be considered.

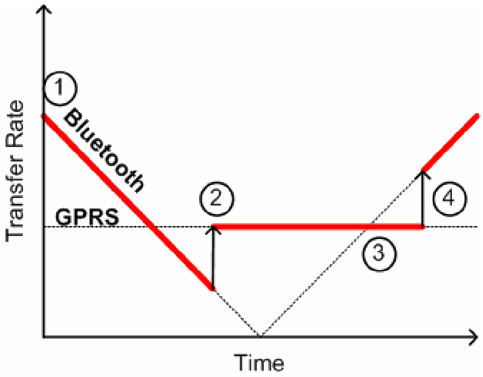


Fig. 2. In stage 1, the Bluetooth connection will continue to be the preferred connection for a certain amount of time even though GPRS has higher transfer rates. In stage 2, GPRS gets a better V_c and a connection switch is performed. In stage 3, Bluetooth gets a better V_c again and eventually in stage 4 the Seamless Connection starts to use Bluetooth again.

3.2 Real Time Connection Switching

The optimal connection process should continuously evaluate the optimal connection using formula (1) for every connection. At the initial stage each connection has assigned an initial value for the transfer rate based on tests performed in section 5, and the optimal connection is chosen. If a connection breaks down the evaluation is executed again setting a of the broken connection to 0 and the transfer will continue using another connection. Transfer rates are not constant but will vary due to changing conditions and numbers of users on the network, physical length from the receiver, interference on a wireless link etc. If the transfer rate decreases so much that another connection has a higher V_c, the application will perform a connection switch. The transfer rate is calculated as the average of recently sent measurements. Figure 2 illustrates this handoff process.

3.3 Synchronized Switching

If a connection between two peers breaks down, both peers need to do the connection switching. When a connection between two peers is established, a synchronization message containing the available connections and their V_c is sent. If a connection switch is known in advance, that is if the optimal connection chooses to switch due to a better alternative being available, all connecting peers are notified so that they can synchronize the switching. If the connection simply breaks down, the receiving peer simply has to try and listen on the sending peer's next best connection alternative from the list.

The Seamless Connection message is to make sure that the peers agree on the connection routines. In this case the messages are wrapped inside a JXTA message.

- **Initiation message.** This message is sent from the sender to the receiver the first time two peers establish contact. The message contains a list of available connections and ranking of the connections. Both peers store the new peer and its preferred connection values. If there are changes to the connection availability at a later time, the peer will again send an initiation message that will overwrite the previous settings.
- **Switch connection message.** A message stating that a connection switch will take place. Both peers will switch to the agreed connection.

4 Design and Implementation

The case study chosen for the evaluation of the framework is the Seamless Connectivity. We have implemented a controller that manages connections with two implementations: GPRS, and Bluetooth connectivity, each with an observer that the controller uses to collect information about the connection. The observer records connection speeds, availability and keeps track of average connection speed. Based on this, the controller uses formula (1) to calculate V_c as described in section 3.

The application starts by initialing the controller and requests to connect to the remote peer. The controller evaluates all implementations it has control over and connects to the optimal connection. The application can then, through simple interfaces, communicate with remote peers.

All data that are sent, is first passed on to the Seamless Connection before it is sent to the remote peer. The Seamless Connection caches the data and continues the transfer using alternative connection if the current connection breaks down.

5 Evaluation

This evaluation consists of two parts; evaluation of the Seamless Connection component and evaluation of the framework.

5.1 Seamless Connection Evaluation

The test environment consists of a smart phone and a laptop communicating via HTTP and Bluetooth. Nokia 6630 with Symbian OS [12] running J2ME MIDP 2.0 is used as the client and a laptop running Windows XP with J2SE 5.0 using BlueCove [13] API for Bluetooth communication is used as a server. The application is communicating using JXTA messages. The J2ME implementation contains source code from the chat application of the JXME project [14] and the JXME Bluetooth project's sample application [15]. The server is based on BenHui's SPP server application [16].

Table 1 shows the initial values of the Seamless Connection. K_c is set to 5 and Bluetooth is the preferred connection. K_c will vary depending on the currency that is being used. The prototype uses Australian dollars.

Table 1. Chosing Otimal Connection

	Bluetooth	**GPRS**
T_i	1,036	13,129
KB/s	20	9
M	1024	1024
C_b	0	0.1
K_c	5	5
a	1	1
V	**52,236**	**126,91**

To evaluate the Seamless Connection, 50 messages are sent from the smart phone to the laptop, waiting one second between receiving one message until sending the next one. During the transfer the smart phone is moved away from the laptop until the Bluetooth connection breaks down (about 10 meters), the message then continues transferring using GPRS. After a while the smart phone is again moved into range of the laptop and the connection switches back to Bluetooth. This mechanism requires a real time evaluation of connections other than the active one.

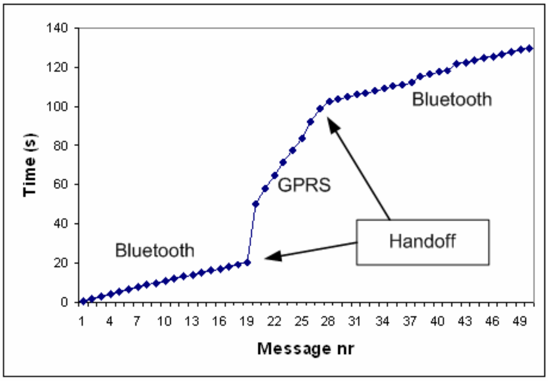


Fig. 3. Test Results. The application starts using Bluetooth but when it gets unavailable it switches to GPRS. When the user gets within Bluetooth range the switch is made back.

5.2 Framework Evaluation

The proposed Dynamic Evaluation Framework will consist of several independent components that are built based on the same basic idea; that there is more than one way of performing a task, and you do not know the optimal method until you have evaluated the options. As described in section 2 we attach observers to the implementations to measure the performance of each implementation. This method requires some custom programming of either the implementation itself or at least the observer has to be custom made for each implementation. The observer for one implementation will be highly reusable for similar implementations.

The framework contains components that continuously evaluate for the optimal implementations, using observers and controllers, and it presents a simple API for the application.A method for adding and removing components in real time is vital if a framework like this should be efficient in real life. A method for deploying components on demand is out of the scope of this article but a context aware deployment [17] will be incorporated in further work.

The hierarchical structure makes it possible to group implementations that are similar. This will minimise the evaluation overhead and it also makes it less complicated to have two slightly different implementations of the same interface. For instance two different routing algorithms in a MANET [18] that performs differently in different scenarios.

6 Conclusion and Future Work

Using the Seamless Connection as an example to evaluate the holistic framework we show that the method of attaching an observer to an implementation is an effective way to monitor components. The controller makes decisions based on reports from the observers and presents a uniform way of connecting to a peer for an application.

This abstracts away implementation specific configuration for the application; it chooses the level of control itself. Each new type of component needs a specific evaluation algorithm based on the parameters available although it is not possible to add new components real time. A hierarchical structure enables applications to have several implementations of interfaces for different scenarios and these implementations are monitored and evaluated real time, ensuring that the optimal solution for every scenario is used.

The framework assumes values used in the evaluation of an implementation are measured real time. These measures can create some unnecessary overhead in scenarios where there is no need to change implementation.

Similar frameworks have been proposed in [7] [19]. They facilitate different aspects of creating generic applications, and a combination of these frameworks with the proposed Dynamic Evaluation Framework can prove to be a good solution.

Continuous evaluation of the optimal connection creates some overhead and will use resources that could be avoided. A scheme for minimizing the evaluation while a connection is connected should be developed.

References

1. Bluetooth website, http://www.bluetooth.com/bluetooth/
2. Infrared Data Association, http://www.irda.org/
3. IEEE 802.11, The Working Group for WLAN Standards. http://standards.ieee.org/announcements/pr_tgp802.11n.html
4. Chen, L.-J., et al., Universal Seamless Handoff Architecture in Wireless Overlay Networks. Technical Report TR040012, UCLA CSD, 2004.
5. JXTA Technology: Creating Connected Communities, 2004, http://www.jxta.org/project/www/docs/JXTA-Exec-Brief-032803.pdf
6. Arora, A., C. Haywood, and K.S. Pabla, JXTA for J2ME– Extending the Reach of Wireless With JXTA Technology, Sun Microsystems, www.**jxta**.org/project/www/docs/JXTA4J2ME.pdf, 2002.
7. Frei, A. and G. Alonso, A dynamic lightweight platform for ad-hoc infrastructures. 2005, http://www.iks.inf.ethz.ch/publications/publications/percom05.html.
8. Jun-Zhao, S., et al., Channel-based connectivity management middleware for seamless integration of heterogeneous wireless networks, in Proceedings of 2005 Symposium on Applications and the Internet (SAINT'05), 2005, p. 213-219.
9. Jun-Zhao, S., et al.,Towards connectivity management adaptability: context awareness in policy representation and end-to-end evaluation algorithm, in Proceedings of the 3rd international conference on Mobile and ubiquitous multimedia. 2004, ACM Press: College Park, Maryland.
10. Harjula, E., et al. Plug-and-play application platform: towards mobile peer-to-peer, in Proceedings of the 3rd international conference on Mobile and ubiquitous multimedia (College Park, Maryland, October 27 - 29). 2004: ACM Press, New York.
11. Ala-Kurikka, J., et al., Empirical aspects on implementing application supernetworking. in Proceedings of NRS/FWCW conference, 2004.
12. Symbian OS. http://www.symbian.com/
13. Project BlueCove, http://sourceforge.net/projects/bluecove/.
14. JXME website, http://jxme.jxta.org/.

15. Bluetooth JXME Project, http://bluetooth-jxme.jxta.org/.
16. BenHui, http://www.benhui.net/.
17. Chantal Taconet, Erik Putrycz, Guy Bernard, Context Aware Deployment for Mobile Users, in Proceeding of 27th Annual International Computer Software and Applications Conference, COMPSAC, 2003, p. 74.
18. Perkins, C.E., et al., Performance comparison of two on-demand routing protocols for ad hoc networks, IEEE Personal Communications, February 2001, p. 28.
19. Kortuem, G., et al., When Peer-to-Peer comes Face-to-Face: Collaborative Peer-to-Peer Computing in Mobile Ad hoc Networks, in Proceedings of First International Conference on Peer-to-Peer Computing (P2P'01), August 27 - 29, 2001. Lingköping, Sweden.

SOAM: An Environment Adaptation Model for the Pervasive Semantic Web

Juan Ignacio Vazquez, Diego López de Ipiña, and Iñigo Sedano

MoreLab - Mobility Research Lab,
University of Deusto,
Avda. Universidades 24, 48007 Bilbao, Spain
{ivazquez, dipina}@eside.deusto.es, isedano@tecnologico.deusto.es

Abstract. Nowadays, there is a major interest in applying Web and Semantic Web techniques for the creation of pervasive computing scenarios, where devices and objects communicate using these technologies. The Web model has largely proved validity both in Internet-wide and intranet scenarios, but it is starting to be applied in personal area networks as a communication and knowledge reasoning system.

In this paper we present SOAM, an experimental model for the creation of pervasive smart objects that use Web and Semantic Web technologies in new ways – resulting in the novel concept of Pervasive Semantic Web – for enabling personal area semantic communication and reasoning processes in order to provide environment adaptation to user preferences.

1 Introduction

The ultimate goal of Ambient Intelligence [1] is to create intelligent spaces to empower users in everyday tasks at home, work, street, vehicle and others. In this vision, environments are proactive perceiving users' surrounding information, often referred as context, and reacting in the appropriate way to facilitate user's activities. In fact, more and more designers are starting to think that the most valuable resource in such environments are not computing power, communication or storage capabilities, but user interaction [2].

Since intelligent objects are not isolated and should not act in their own, some kind of infrastructure, communication and reasoning model must be provided to guarantee that coordination and organisation activities among those objects are performed to facilitate users' experience.

Traditional Web technologies, such as HTTP, HTML and XML, have been used for providing presentation and control mechanisms in pervasive computing environments, creating a sort of Personal Area Web for interacting with devices (e.g. UPnP [8]). We think that these models can be augmented with Semantic Web technologies to provide reasoning processes at the devices and at the environment itself.

In this way, full intelligent environments can be created where reasoning processes are automatically performed, communicated and agreed upon between

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 108–117, 2006.

intelligent objects, based on users' context perceptions and other available inputs. The goal is to adapt the environment without user intervention [9] [11] (maximizing intelligence by minimizing explicit user interaction).

In this paper, we present SOAM – Smart Objects Awareness and Adaptation Model –, a proposal joining the forces of well-known Web-based communication mechanisms with intelligent capabilities provided by Semantic Web technologies in a structured manner, so that smart objects, or *smobjects* the term we coined, can be easily designed to create user-aware adaptive intelligent spaces following simple principles. SOAM is a preferences-based environment adaptation model in which part of user's context is built up by semantic preferences about environmental conditions. These preferences lead to adaptation on smobjects without explicit user intervention, yet allowing automatic reasoning over those preferences.

There are a number of technologies and initiatives related to this field that constituted the background for our work, such as Universal Plug and Play (UPnP) [8], Task Computing [4] and SOUPA (Standard Ontology for Ubiquitous and Pervasive Applications) [3].

In section 2, the smobject concept is detailed as well as the exchanged information structures, while in section 3 the SOAM adaptation model and involved entities are described. Finally, in section 4 some future research directions are given.

2 Pervasive Smart Objects

2.1 The Pervasive Semantic Web Vision

We coin the term *Pervasive Semantic Web* to designate the result of applying Semantic Web technologies to Pervasive Computing scenarios in order to perform reasoning processes. The main representatives of those technologies are RDF (Resource Description Framework) [6] and OWL(Ontology Web Language) [7].

These scenarios are populated by different kinds of devices with a number of capabilities such as temperature sensing, video capturing, door opening, and so on. Our strategy is based on using ontologies to represent knowledge about different domains, so that we use appropriate ontologies for temperature, physical access control, location and so forth.

In our vision, we pretend to create some kind of Personal Area Web, where devices are interconnected, hosting knowledge about environmental perceived conditions and using references to link resources inside and outside this space. This vision determines the creation of a new type of logical environment in ubiquitous computing scenarios: a personal area semantic web with information flows back and forth among communicating devices, sharing their knowledge about users inside the environment and coordinating their tasks via distributed reasoning procedures in order to provide an ambient intelligence experience.

This point of view about future living and working environments is shared with other research groups that provided similar viewpoints [5], but until now there are no practical results or architectures developed and tested.

SOAM – Smart Object Awareness and Adaptation Model – is intended to fill this gap by providing a comprehensive architecture, easily replicable, to test automatic environment adaptation scenarios by applying semantic annotation techniques.

2.2 Smobjects: Smart Objects

In SOAM a major entity of the architecture is what we denominate *smobject* as a short for "smart object". A smobject is an agent in the form of a piece of software, representing an intelligent device, several devices or event part of a device. Smobjects capture a subset of environmental conditions, provide that perceived context information under request, and act upon the same or other subset of environmental conditions in order to modify them and adapt them as needed. Smobjects need to access built-in sensors, effectors, communication ports, maybe storage facilities if available on the device, and so forth.

The real strength of the smobject-based agent architecture in SOAM is that standardizes the way sensors and effectors information is represented and accessed. Smobjects manage three different types of information structures that illustrate the functions a smobject can carry out:

- **Context Information**: smobjects provide information about perceived state of the environment via semantically annotated data under request. Context Information is gathered through built-in sensors in the bounded device and provided by the smobject to requesting parties.
- **Capabilities**: a smobject is capable of perceiving only some concrete environmental conditions depending on the bounded-device built-in sensors, and it is also capable of operating over some (same or other) conditions depending on the bounded-device built-in effectors. Perception and operation capabilities are provided by the smobject to requesting parties.
- **Constraints**: smobjects can be influenced by other entities using some data constructions called constraints, which declare valid ranges on the desired state of the environment, so that the smobject is in charge of driving adaptation honouring them. Smobject's behaviour is defined by active constraints, which represent existing influences over the smobject, and have a limited lifespan. Smobjects can provide information about their active constraints to requesting parties, as well as accept constraints from other entities that desire to influence the smobject's behaviour.

In SOAM, these data entities are exchanged through smobject standard communication interfaces as shown in the figure 1. We can also notice how the smobject interacts with the host device, using their built-in sensors and effectors.

Capabilities and Constraints are represented and exchanged in XML using structures declared in a grammar called SOAM Datatypes XML Schema, while Context Information is represented using RDF and domain ontologies honouring the OWL specification. Standard HTTP is used for transport and negotiation purposes between other entities and the smobjects in order to retrieve and store

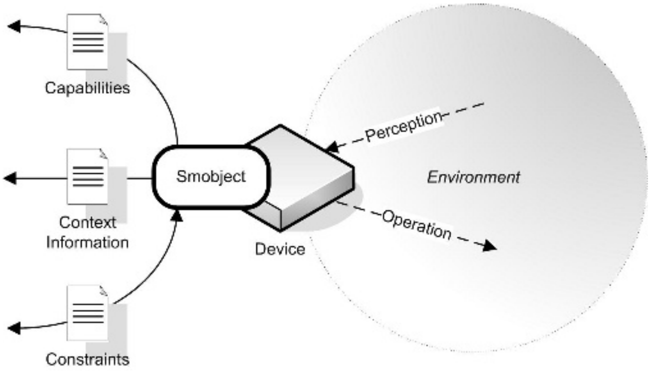


Fig. 1. Smobject communication interfaces

these information structures in SOAM (HTTPS could be used to facilitate a secure communication channel, providing every smobject has a valid and trustable X.509 certificate).

2.3 Context Information

Context Information is probably the most important data a smobject can provide. Context Information is constructed using RDF, serialized in the form of XML. It conveys perceived information captured through device's sensors, annotated via RDF and OWL. Captured data semantics is highly knowledge domain dependent, for example temperature measures, an item location or an elevator's present position.

Since devices are specialized in domains (TV, temperature control system, light), smobjects act as control processes deeply associated to the concrete device to act upon built-in sensors and effectors and programmed to semantically annotate the perceived data using the most appropriate and standard ontology for that purpose. It is up to devices' and smobjects' designers to select suitable ontologies among available ones.

An example of a Context Information message conveying knowledge about luminance is shown if figure 2. Probably, the smobject is installed in a lighting device called `light1`, and it provides information about `light1`'s state upon request (luminance, light color, ...).

As we can notice, a smobject normally does not only provide information about perceptions obtained by sensors, but also the device identification and type, that is, the full semantic description of available data. This annotation is particularly useful to automate processes depending on device identification, type or other device parameters.

2.4 Capabilities

A smobject can exhibit perception capabilities on some domains and operation capabilities on the same or different domains. Perception capabilities represent

```
<rdf:Description rdf:about="urn:uuid:light1">
  <lit:luminance rdf:datatype="http://www.w3.org/2001/XMLSchema#int">
    30
  </lit:luminance>
  <lit:color rdf:resource="http://www.awareit.com/onto/light#White"/>
  <rdf:type rdf:resource="http://www.awareit.com/onto/light#Light"/>
</rdf:Description>
```

Fig. 2. An example Information structure provided by a smobject

sensing mechanisms the smobject is able to access on the host device about some domains (for example, lighting conditions), while operation capabilities represent control mechanisms the smobject features about some domains.

For example, a light sensor has perception capabilities about the “lighting domain” in a room, while a switch has operation capabilities about the “lighting domain” (via a lamp o light bulb). Usually, both devices would be modeled using the same “lighting control system” smobject, thus featuring at the same time perception and operation capabilities about the “lighting domain”.

Capabilities are generally not only bounded to a knowledge domain, but also to concrete elements to which the information is related. For instance, a smobject can perceive lighting conditions, but only those related to `light1`. Or maybe, the electronic thermometer smobject measures the existing temperature, but only in `room1`.

Of course, some smobjects can measure conditions related to undefined actors, or unbounded at all. The SOAM Datatypes XML Schema defines data structures to declare perception and operation capabilities, using even wildcards to denote the “any” concept.

Figure 3 is an example of the capabilities file of the previous lighting smobject with both perception and operation capabilities on a concrete light.

Basically, this document means that the smobject can perceive the state of `light1` in the “light” domain (with all the predicates included in the ontology) and it can also adapt `light1` dynamically in the same domain.

```
<capabilitiesCollection>
  <perceptionCapability id="urn:uuid:light1_pcap1">
    <subject resource="urn:uuid:light1"/>
    <ontology resource="http://www.awareit.com/onto/light"/>
  </perceptionCapability>
  <operationCapability id="urn:uuid:light1_ocap1">
    <subject resource="urn:uuid:light1"/>
    <ontology resource="http://www.awareit.com/onto/light"/>
  </operationCapability>
</capabilitiesCollection>
```

Fig. 3. An example Capabilities structure provided by a smobject

```
<constraintsCollection>
  <constraint expires="PT1M"
      subject="urn:uuid:light1"
      predicate="http://www.awareit.com/onto/light#luminance">
    <objectLiteral datatype="http://www.w3.org/2001/XMLSchema#int">
      10
    </objectLiteral>
  </constraint>
  <constraint expires="PT1M"
      subject="urn:uuid:light1"
      predicate="http://www.awareit.com/onto/light#color">
    <objectResource ref="http://www.awareit.com/onto/light#Yellow"/>
  </constraint>
</constraintsCollection>
```

Fig. 4. An example Constraints information provided by a smobject.

2.5 Constraints

Smobjects receive requests to perform environment adaptation through effectors. These requests come in the form of Constraints, represented by statement patterns in the desired behaviour. A smobject can receive a number of this kind of constraints over the time, so its behaviour is influenced and driven by them. In fact, a smobject is in charge of managing the active Constraints and trying to perform in such a way that Constraints are honoured.

The SOAM Datatypes XML Schema provides a way to represent and exchange constraints. Those Constraints are generated by initial configuration settings and/or adaptation requests sent by other actors.

Figure 4 illustrates an example of active Constraints on `light1`.

The previous Constraint could be read as "*light1 must have a luminance of 10 and yellow color*"

Constraints are the unique out of the three data entities (Context Information, Capabilities and Constraints) that can be also injected into smobjects and not only requested from them. As explained previously, in SOAM, any actor can retrieve Constraints from the smobject to find out how its behaviour is being driven, but also existing actors can send Constraints to the smobject to constrain its behaviour and have the environment conveniently adapted.

Since HTTP messages are used in SOAM to negotiate information exchange, HTTP Basic Authentication [10] or other standard web mechanisms can be used for identification and authentication purposes if needed.

3 Environment Adaptation

3.1 Adaptation Profiles

The goal of SOAM is to achieve a comprehensive model for automatic adaptation of the environment to user preferences, needs and behavioural patterns. As shown, smobjects are the entities in charge of performing the final operations to achieve adaptation.

Adaptation Profiles are the information elements that conveys user's adaptation requirements that eventually drive smobjects behaviour. Adaptation Profiles are stored and exchanged with the environment via the user's personal device, which contains an Adaptation User-Agent in charge of negotiating Adaptation Profiles with surrounding entities as explained below.

An Adaptation Profile is a conditional preference or environment adaptation requirement that contains two different sections:

- **Preconditions**: represent existing requirements about the environment's present state, that must be met for the Adaptation Profile to activate. It makes the adaptation to have a conditional nature. Often, adaptation requirements are not fixed, e.g. a user does not need his preferred temperature to be always 22^{o}C, but maybe only when he is at the car.
- **Postconditions**: represent desired patterns in the environment's future state that must be met for the adaptation to be considered as honoured. Postconditions eventually generate constraints.

Variable substitution in Adaptation Profiles is possible to allow postcondition elements to be bounded to precondition elements as shown in figure 5.

This Adaptation Profile can be read as "*whatever the location Alice is in with an ambient light, that ambient light should have a luminance of 90*", which is a very simple but powerful mechanism to force every location's lights to adjust automatically as Alice gets in.

```
<adaptationProfile id="urn:uuid:prof1" expires="PT2M">
  <variable id="x"/>
  <variable id="y"/>
  <precondition subject="urn:uuid:Alice"
      predicate="http://www.awareit.com/onto/location#isLocatedIn">
    <objectVariable ref="x"/>
  </precondition>
  <precondition subject="x"
      predicate="http://www.awareit.com/onto/light#hasAmbientLight">
    <objectVariable ref="y"/>
  </precondition>
  <postcondition subject="y"
      predicate="http://www.awareit.com/onto/light#luminance">
    <objectLiteral datatype="http://www.w3.org/2001/XMLSchema#int">
      90
    </objectLiteral>
  </postcondition>
</adaptationProfile>
```

Fig. 5. An example Adaptation Profile with bounded variables

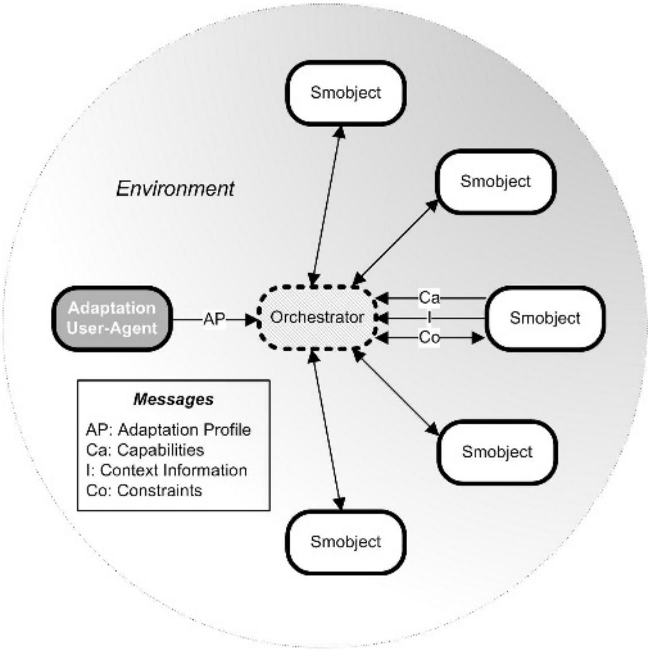


Fig. 6. Diagram illustrating SOAM architecture

3.2 Other SOAM Entities

Despite smobjects play a fundamental role in the SOAM architecture, they just act as intermediates with the associated device to fulfil adaptation requests. There are some other entities needed in SOAM in charge of generating those requests on behalf of the user and instructing smobjects to adapt the environment in a coordinated way:

- **Adaptation User-Agent**: a piece of software acting on behalf of a user that is aware of the user's Adaptation Profiles and negotiates with surrounding Orchestrators the adaptation process to exchange those profiles.
- **Orchestrator**: an entity that perceives and orchestrates existing smobjects in the environment to perform the adaptation process following Adaptation Profiles. Orchestrators feature semantic information reasoning and a rule engine in order to generate Constraints from Context Information and Adaptation Profiles.

Adaptation User-Agents act generally on behalf of a user, silently starting the process of adapting the environment by finding an available Orchestrator to which they send user's Adaptation Profiles.

4 Conclusions and Future Work

SOAM is an effort to create a comprehensive model for automatic environment adaptation to user's preferences, based on existing well-proven technologies such as SSDP, HTTP and XML, as well as the Semantic Web (RDF, OWL) for knowledge representation and reasoning. SOAM is based in a special kind of pervasive agents called smobjects that interface with real devices via a standard interface for exchanging information.

Our current prototype implementation is based in a single board computer in the role of Orchestrator, using Jena libraries for semantic information processing, and ARM9 embedded processors (UNC20) for smobjects. Adaptation User-Agents can be implemented in PocketPC o cellular phones. Some experimental scenarios, related to home and intelligent workplace environments are being created to test SOAM feasibility and capabilities. SOAM can take advantage of standard ontologies, such as SOUPA, for concrete domain knowledge representation.

SOAM illustrates the possibilities of the new paradigm emerging from the joint forces of the Web and Semantic Web technologies applied to Pervasive Computing scenarios, creating Pervasive Semantic Webs everywhere and augmenting intelligence in environments.

There are some open research issues related to SOAM architecture that still need to be studied such as conflict resolution with multiple users' disjoint requirements, usage of standard orchestration languages, or the possibility of removing the Orchestrator element of the architecture to create true decentralized choreography among smobjects.

Acknowledgements

This work has been partially supported by the Department of Industry, Commerce and Tourism of the Basque Government under the SAIOTEK grant S-OD04UD02, and the Cathedra of Telefonica Moviles at Deusto University, Bilbao, Spain.

References

1. K. Ducatel, M. Bogdanowicz, F. Scapolo, J. Leijten and J-C. Burgelman. *Scenarios for Ambient Intelligence in 2010. Final Report.* IST Advisory Group. EC (2001).
2. Project Aura. http://www.cs.cmu.edu/ aura/
3. H. Chen et al. *SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications.* Proceedings of Mobiquitous 2004: International Conference on Mobile and Ubiquitous Systems: Networking and Services, Boston, USA (2004).
4. R. Masuoka and Y. Labrou. *Task Computing - Semantic-web enabled, user-driven, interactive environments.* WWW Based Communities For Knowledge Presentation, Sharing, Mining and Protection (The PSMP workshop) within CIC 2003, Las Vegas, USA (2003)
5. Ora Lassila. *Using the Semantic Web in Mobile and Ubiquitous Computing.* Proceedings of the 1st IFIP WG12.5 Working Conference on Industrial Applications of Semantic Web, pp. 19-25. Springer (2005).
6. World Wide Web Consortium. *RDF Primer. W3C Recommendation.* World Wide Web Consortium (2004).

7. World Wide Web Consortium. *OWL Web Ontology Language Semantics and Abstract Syntax. W3C Recommendation.* World Wide Web Consortium (2004).
8. UPnP Forum. *UPnP Device Architecture1.0.* UPnP Forum (2003).
9. J. I. Vazquez and D. Lopez de Ipiña. *An Interaction Model for Passively Influencing the Environment.* Adjunct Proceedings of the 2nd European Symposium on Ambient Intelligence, Eindhoven, The Netherlands (2004).
10. J. Franks et al. *RFC 2617: HTTP Authentication: Basic and Digest Access Authentication.* IETF RFC (1999).
11. J. I. Vazquez and D. Lopez de Ipina. *A language for expressing user-context preferences in the web.* WWW 2005: Special interest Tracks and Posters of the 14th international Conference on World Wide Web (Chiba, Japan) pp. 904-905. ACM Press (2005).

Implementing the MPEG-21 Adaptation Quality of Service in Dynamic Environments

Marios C. Angelides[1], Anastasis A. Sofokleous[1], and Christos N. Schizas[2]

[1] Brunel University, Uxbridge, Middlesex, UB8 3PH, UK
{marios.angelides, anastasis.sofokleous}@brunel.ac.uk
[2] University of Cyprus, Nicosia, Cyprus
schizas@ucy.ac.cy

Abstract. MPEG-21 embeds the AQoS schema tool which describes and associates conceptually major utilities required during the adaptation process, for example, adaptation operators, constraints and qualities. Defining an AQoS model, i.e. possible adaptation operators, predictable constraints and qualities that will be included in an instance of AQoS which accompanies a media file, and consequently initialising its values is a complex process. In this paper, we present our conceptual model design for implementing the AQoS.

1 Introduction

Multimedia content adaptation aims to personalise multimedia content before delivery to individual users. Adaptation is necessary in order to make the content universally accessible (Universal Multimedia Access) since different networks (of varying capabilities with regards to bandwidth, data loss, flow control, error control, etc.), devices (with varying memory, resolution, refresh rates, etc.) and users (with varying individual preferences, usage history, etc.) exhibit different constraints and requirements in terms of quality of service, communication, processing, presentation[1][2]. Content adaptation involves a series of processes, such as selection of either a single or a combination of adaptation operations and population of their parametric values. However, the possible combinations of such operations are enormous since each combination will need to be optimised in consideration of the network and device characteristics. Each combination will most certainly yield a different level of satisfaction for each user. However, before searching for optimum adaptation solutions it is necessary to define the search domains (i.e. which adaptation operations will be included and what possible values they will have), constraints (i.e. which adaptation operations will be excluded as being non-valid) and the evaluation utilities (i.e. which evaluation factor will provide one or more valid solutions). Whilst MPEG-21's AQoS tool caters for all the above, it, nevertheless, does not prescribe how AQoS should be build during each adaptation exercise.

The objective of this paper is to present our design of a flexible AQoS model for video adaptation and our algorithms for automatic AQoS implementation. Section 2 gives an overview of R&D in the area followed, in section 3, by our approach to implementation of AQoS. In section 4 we test our AQoS implementation. Finally, the paper concludes with our view of possible future R&D.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 118 – 127, 2006.

2 Research Overview

Many researchers and research practitioners have investigated various methods for achieving content adaptability, for example, transforming a media file by changing its format, scale, rate, or quality [3]. In many such cases, either a manual or an automatic decision-taking process is deployed to choose the right set of parameters for the adaptation operation(s). [4] and [5] emphasise the density of the adaptation process by defining a multidimensional space of "adaptation operations, resources and utilities (ARU) spaces". An adaptation operation may include a number of adaptation methods, such as transcoding and summarisation, and their parameter values. Each adaptation operation in each dimension is represented by a set of co-ordinates and a point of reference in the multidimensional space represents a combination of operations from different dimensions. The resource space includes the resources that are available to the adaptation process, such as a device's display resolution, colour depth and bandwidth. This kind of information can be used to describe the limitation and optimization constraints during the adaptation process. The utility space describes user satisfaction values. Collectively, the ARU spaces define a complete framework for adaptation decision making since it can formulate a constrained optimisation problem involving algebraic variables from the three spaces independently of what the variables represent [6]. The MPEG-21 framework suggests that during digital item consumption, content adaptation can be utilised through a resource adaptation engine and/or a descriptor adaptation engine [7, 8].

2.1 MPEG-21's AQoS Tool

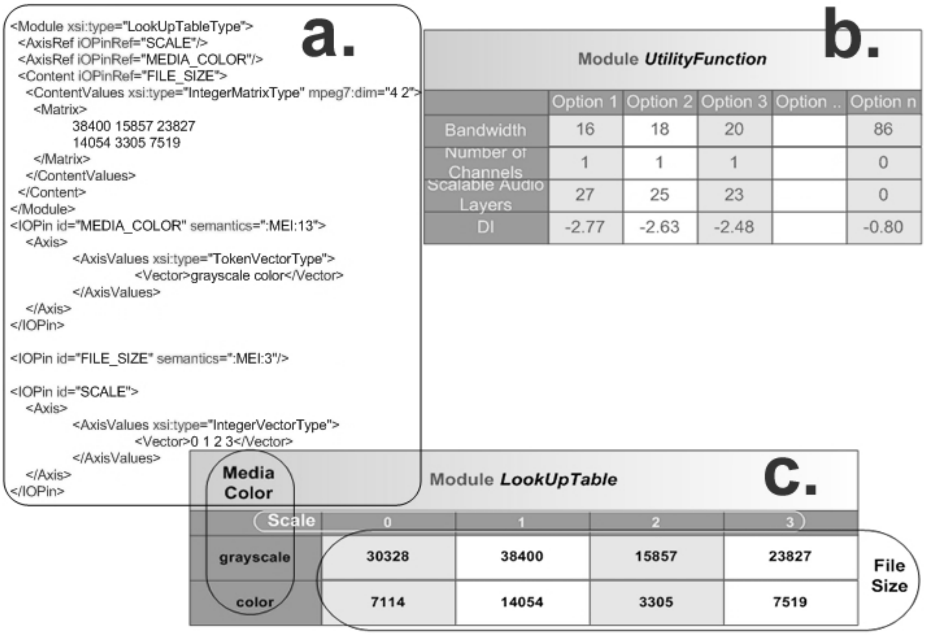
The *AQoS* tool describes the relationship between Quality of Service constraints (e.g., on network bandwidth or a terminal's computational capabilities), possible adaptation operations that may satisfy these constraints, and associated media resource qualities that result from adaptation. This set of tools provides the means to trade-off these parameters with respect to quality so that an adaptation strategy can be formulated and optimal adaptation decisions can be made in constrained environments. AQoS has been designed in a modular way, and therefore an instance of AQoS can be constructed using a number of interconnected modules of *UtilityFunction*, *LookUpTable* or *StackFunction* (table 1).

Each module allows different interpolation. Figure 1a presents a *Lookuptable* example for a media file as defined by the *AQoS* schema, in which a selected value for the *filesize* axis will define the value of the other two values (media colour and

Table 1. AQoS Module Types

UtilityFunction	Distribution of three key factors involved in media resource adaptation: • constraint (e.g., bandwidth, power, display resolution) • adaptation operation (e.g., frame dropping, spatial size reduction) • utility (e.g., objective or subjective quality, such as PSNR, Distortion Index (DI))
LookUpTable	Matrix representation of data and their relationship (axes and content)
StackFunction	Mathematical relationships between the variables (IOPins)

scale). In figure 1c we attempt to provide a logical representation of the 1a *lookuptable* example. For a *utilityfunction*, a logical representation is given in figure 1b. Globally a variable (IOPin) must have the same value. The *AQoS* is generated for each media file (since it provides different values for each media file) in a media resource server and is delivered along with the associated media resource to an adaptation engine located on a network proxy or a terminal. Using semantics and other referencing mechanisms, the *AQoS* can make reference to its internal variables and also to other external variables (e.g. to user characteristic –*UED*) and be referenced by other tools (e.g. by a constraint in the *UCD*)[1]. The Universal Constraint Description tool (UCD) describes limitation constraints and optimisation objectives on the AQoS parts that affect the adaptation decisions (e.g. on an adaptation operation, on a quality, or constraint). This tool is indented to be used by a consumer or a provider so as to specify their specific constraints and objectives for each digital item during the adaptation. Using references and semantic the UCD can make reference to user environment description data, AQoS data and video metadata.



```
<Module xsi:type="LookUpTableType">
  <AxisRef iOPinRef="SCALE"/>
  <AxisRef iOPinRef="MEDIA_COLOR"/>
  <Content iOPinRef="FILE_SIZE">
    <ContentValues xsi:type="IntegerMatrixType" mpeg7:dim="4 2">
      <Matrix>
            38400 15857 23827
            14054 3305 7519
      </Matrix>
    </ContentValues>
  </Content>
</Module>
<IOPin id="MEDIA_COLOR" semantics=":MEI:13">
    <Axis>
            <AxisValues xsi:type="TokenVectorType">
                        <Vector>grayscale color</Vector>
            </AxisValues>
    </Axis>
</IOPin>

<IOPin id="FILE_SIZE" semantics=":MEI:3"/>

<IOPin id="SCALE">
    <Axis>
            <AxisValues xsi:type="IntegerVectorType">
                        <Vector>0 1 2 3</Vector>
            </AxisValues>
    </Axis>
</IOPin>
```

Module *UtilityFunction*

	Option 1	Option 2	Option 3	Option ..	Option n
Bandwidth	16	18	20		86
Number of Channels	1	1	1		0
Scalable Audio Layers	27	25	23		0
DI	-2.77	-2.63	-2.48		-0.80

Module *LookUpTable*

Media Color / Scale	0	1	2	3
grayscale	30328	38400	15857	23827
color	7114	14054	3305	7519

Fig. 1. AQoS example

3 AQoS Implementation

The process for determining the best AQoS model for a digital item is fuzzy since the possible combinations of all the factors involved is enormous and each user might request content based on different type of adaptation requirements. For example, whilst the resource space might be constrained by bandwidth, audio bit rate, video bit rate, resolution, etc. at the same time audio bit rate adjustment may also be an adaptation operation alongside transcoding operations such as resolution dropping, FD-CD, audio channel configuration, format conversion, etc. An AQoS instance is a space of possible adaptation operations, constraints and user preferences. The definition of the *AQoS* for a media file should capture the following (i) a suitable *AQoS* model, i.e. adaptation operations, resource constraints and user preferences, and (ii) a population method for the *AQoS* final file. For example, if resolution dropping is such an adaptation operation that converts the original media into a degraded version observing resource limitations (e.g. bandwidth) or user's preferences then the operation would require values for *bandwidth*, the *frame-width* and *frame-height* and the PSNR of a media file to be generated.

Table 2 presents a conceptual and flexible AQoS model that may be used during audio-visual content dynamic adaptation. Whilst this instance of the model uses only spatial transcoding (resolution dropping) and colour transcoding (colour depth, Colour/greyscale) techniques, it can be easily expanded for temporal transcoding (adjusting the frame rate or enforcing frame dropping), code transcoding (change compression/ coding standard) and semantic or object transcoding (different transcoding for each object that has semantically different value to the user).

Table 2. AQoS Proposed Model

Type	Description	Content
Video Utility function	video track of segmentation	*filesize*(constraint), *video-bit-rate* (adaptation), *scale* or *resolution dropping* (adaptation), *media-colour* (adaptation)
Audio Utility Function	audio track of segmentation	*file-size* (constraint), *audio-bit-rate/ number-of-audio-channels* (adaptation)
Lookup table	frame width/height for each scale	*Scale* (scale=1 → w=800/h=600)
Lookup table	possible channel configuration	*Channel-Configuration*
Stack Function	total file size = sum of audio and video file size	*total-file-size*
Stack Function	required bandwidth	*Bandwidth*

Generating AQoS model values is a complex process. Assuming an AQoS model (e.g. the table above), there are not algorithms for calculating automatically the values without exercising some degree of selection. Testing possible values, initially, is usually necessary since AQoS usually includes many adaptation operations with each adaptation operation using its own "prediction" algorithms. In this paper we propose a multiphase automatic process for implementing the AQoS, and consequently for generating the model values with minimum human effort. The basic phases are as follows:

- Phase 1: Video Annotation
- Phase 2: Video Segmentation
- Phase 3: Audio and Video Track Separation
- Phase 4: AV Track Variation Generation
- Phase 5: AQoS Switch Values Estimation
- Phase 6: AQoS Construction

PHASE 1: Video Annotation

Annotating the media file is inevitable because it will provide useful metadata about the media file under analysis and it will also assist in creating the AQoS. The MPEG-7 file describes the video file syntactically and semantically. Consider, for instance, an MPEG-4 sport video clip which illustrates a soccer player kicking the ball. Its MPEG-7 file could include metadata on the video format, the event (e.g. date, time, occasion, location, player name, and playing field name), the objects of interest (e.g. player, manager, pitch, etc.). Video segmentation is usually driven by competing factors such as visual similarity, time locality, coherence, colour, motion, etc. Processing each video segment individually requires description of its characteristics independently, which in turn will require different adaptation options, e.g. no audio for the first shot, high video quality for the second shot, etc.

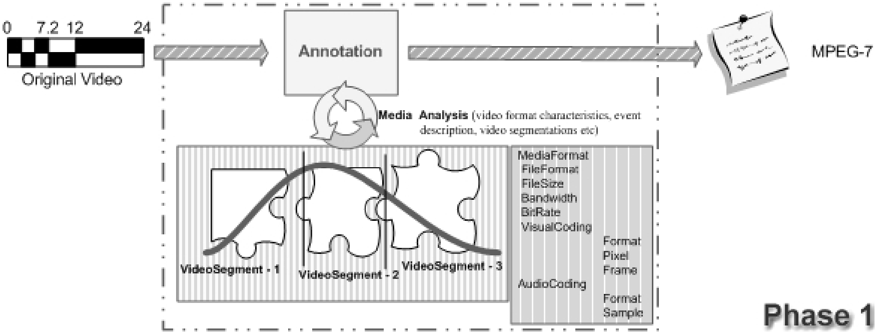


Fig. 2. Video Annotation

PHASE 2: Video Segmentation

A video segmentation algorithm identifies the shot/scene boundaries of media stream (phase 1). An MPEG-7 file records all metadata about each segment of the stream. Whilst phase 1 is only executed once for the entire video stream, phases 3 and 4 repeat for every video segment.

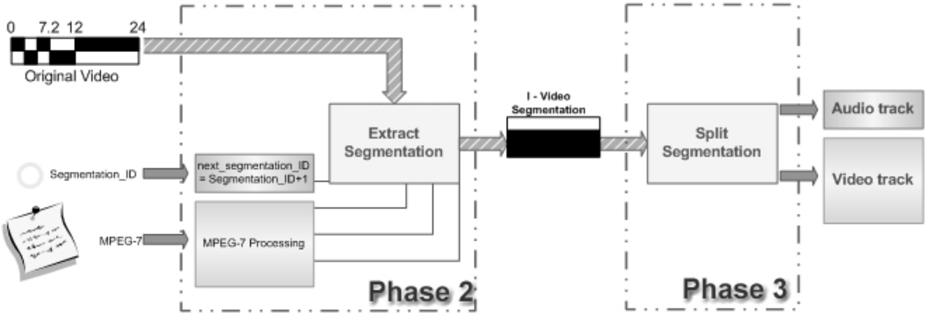


Fig. 3. (a) Extracting the video segments (b) audio and video track separation

PHASE 3: Audio and Video Track Separation

The video segments are separated into Audio and Video tracks.

PHASE 4: AV Track Variation Generation

The purpose of this phase is to generate different variations of the separated AV tracks using a predefined AQoS model like that of table 2. In the context of the AQoS model of table 2, the maximum resolution will be the video original resolution (scale 0: 1024x768). Keeping the original aspect ratio of 4:3, three more degraded scales are generated, i.e. scale 1:800x600, scale 2:600x450 and scale 3:400x300), which are used in the first *lookuptable* (scale/resolution). Likewise, the domain values for the other adaptation operation variables (IOPins) are generated, i.e. video bit rate, colour or greyscale. A media generator process, loads all possible variation jobs so as to calculate them offline and thus discover depended variables, i.e. file size, PSNR etc.

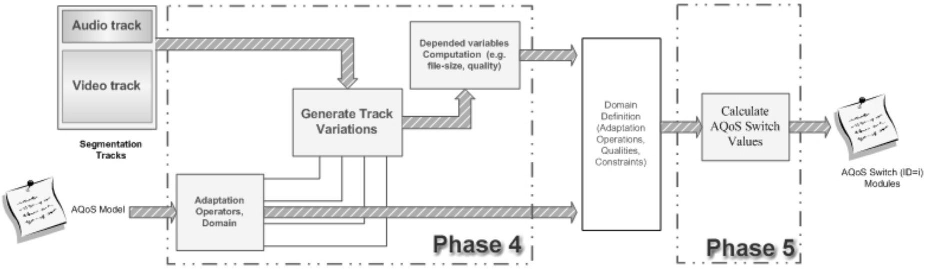

Audio track
Video track
Segmentation Tracks
AQoS Model
Adaptation Operators, Domain
Generate Track Variations
Depended variables Computation (e.g. file-size, quality)
Phase 4
Domain Definition (Adaptation Operations, Qualities, Constraints)
Calculate AQoS Switch Values
Phase 5
AQoS Switch (ID=i) Modules

Fig. 4. (a) AV Track Variation Generation (b) AQOS Switch modules generation

PHASE 5: AQoS Switch Values Estimation

This phase aims to construct the individual AQoS parts, by using the values calculated in phase 4. We use the AQoS Switch mechanism for describing independently each of the video segments of the original video stream, since the resource requirements may vary across segments. Thus, by using the AQoS switch for each video segment, we are able to define in a greater detail the values of adaptation operations, constraints and user preferences. Universal values, such as video scales, can be described outside AQoS switch sections, specifically in the global area of AQoS. For instance, if the AQoS model of table 2 is used, each switch section would have two utility functions (one for video and one for audio) and consequently, the global AQoS would include the lookup tables and stack function modules. For each segment, the possible adaptations are:

$$\#Video\ Segment\ Adaptations = \#Audio\ adaptation\ \ x\ \ \#Video\ adaptation$$

Separating the audio and video tracks enables a minimum level of variation production and value calculation since:

$$\#Segment\ offline\ calculations = \#Audio\ adaptation + \#Video\ adaptation$$

Thus by generating $m + n$ variations we are able to provide $m\ x\ n$ adaptations.

PHASE 6: AQoS Construction

Phase 6 is the last step of the process and assumes a successful execution of all previous phases. In this step, each AQoS switch is integrated into one AQoS which includes global structures. The AQoS model used phase 4 is being used as a template for generating the final AQoS for the original media file. For instance, although the AQoS model defines the video scale module (lookup table, table 2) with 4 or more scales, it does not state what the exact resolution for each scale is. The maximum width and height depend on the original video and are, therefore, recorded only in the final AQoS. The possible adaptations for the entire media stream are:

$$\#\mathit{Total\ Adaptations} = \#\mathit{Segment\ Adaptation}_1\ x \ldots x\ \#\mathit{Segment\ Adaptation}_n$$

And the offline processing is estimated as follows.

$$\#\mathit{Total\ offline\ calculations} = \#\mathit{Segment\ offline\ calculations}_1 + \ldots + \#\mathit{Segment\ offline\ calculations}_n$$

Therefore, by generating $m + n$ variations for the k video segmentations we are able to provide $(m\ x\ n)^k$ adaptations.

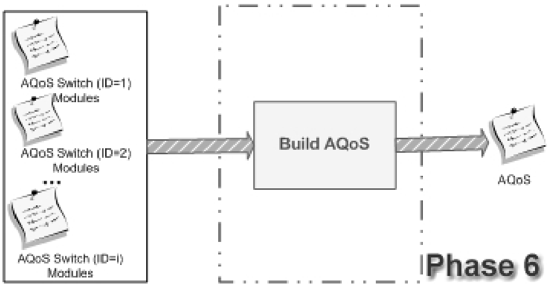


Fig. 5. Final AQoS Construction

4 Phase Implementation Using Java and Java Media Framework

In order to demonstrate the applicability of the proposed multiphase process, we have implemented and tested a multilayer architecture consisting of various video and data processing utilities. The open architecture can be easily expanded to include additional modules, such as video object detection, tracking, and extraction. For the first phase, we have implemented an MPEG-7 annotation tool which is able to read a video file and display its information (i.e. frames as thumbnails, frame number, total frames, etc.) so a user can easily assign the video segment boundaries (see figure 6a). General video information such as original video resolution and average video bit rate, and also segmentation boundaries are recorded in the MPEG-7 file which is the input to phase 2. We have also implemented a number of MPEG-7 and MPEG-21 I/O libraries. The I/O libraries consist of many modules that are responsible for loading data into JAVA structures, providing processing mechanisms (search, insertion, deletion). These are utilised by other modules, such as the AV segment extraction.

Phase 2 utilizes the MPEG-7 I/O library for loading the MPEG-7 file and furthermore it executes a video extraction algorithm for each video stream with the initial parameters being the video file URL, start frame number and end frame number. Although the particular extraction algorithm (cut and display) and also the audio/video split algorithm used in phase 3 use the Java Media Framework, JMF limitations forced the use of the FFmpeg [9] for phase 3. FFmpeg is a very fast video and audio converter which is able to support various transcoding operations, file formats and protocols as input. However, FFmpeg is programmed in the ISO C90 language and therefore we had to implement a java handler (java external executer) for passing the transcoding options and their parametric values in phase 4. The FFmpeg output is redirected through Java Output Redirector module to the FFmpeg Output Transformation module, which cleans the output data (i.e. results, average PSNR, audio file-size, video file-size etc.) and creates the AQoS switch parts for phase 5. Finally the AQoS file is constructed utilizing the AQoS MPEG-21 library functions.

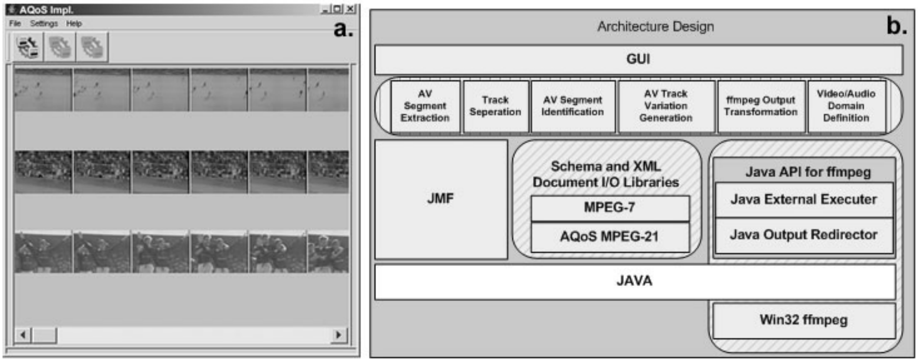
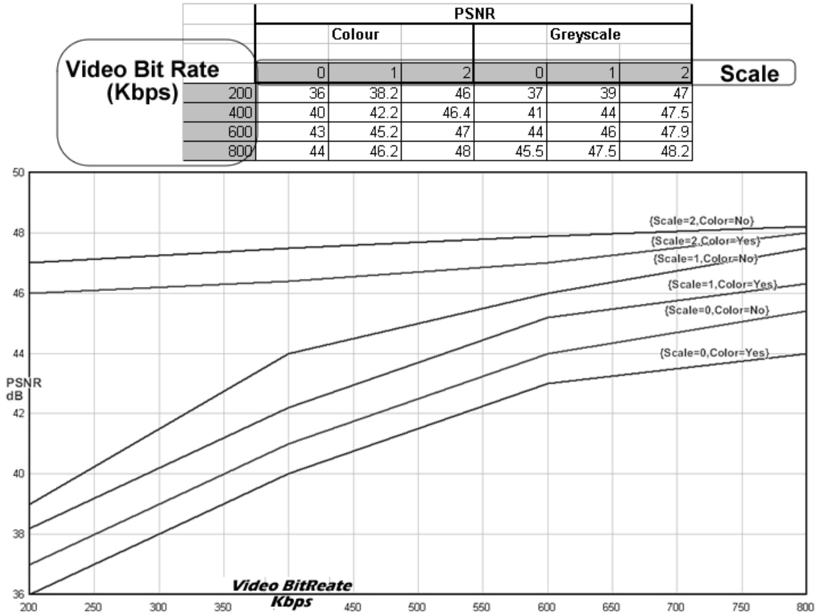


Fig. 6. (a) Application GUI – Video Segmentation (b) Abstract Architecture

5 Assessing the AQoS in a Sample Dynamic Environment

Methods of AV quality assessment are classified into two main categories: subjective assessment (which uses human viewers and audience) and objective assessment (which uses electrical measurements). Subjective quality measurement is related with a number of complex entities of the human AV system such as brain and eyes and ears. Although, the opinion of AV quality is influenced by human factors, how clearly parts of the scene can be seen and whether motion appears natural and ‘smooth’, a viewer’s opinion of ‘quality’ is also affected by other factors such as the viewing environment, the observer’s state of mind and the extent to which the observer interacts with the visual scene. In addition, a user carrying out a specific job that requires attention on part of a visual scene will have quite different needs for ‘good’ quality than a user who is passively watching a movie [10]. Another way for measuring the quality of video relies heavily on objective quality measures. Two conventional objective methods followed for measuring the video compression quality are the Peak Signal to Noise Ratio (PSNR) and Peak Error Calculation. The limitations of the PSNR metric have led to many efforts to develop more sophisticated measures that approximate the response of ‘real’ human observers. The DCT-based video quality evaluation and evaluation based on Spatial and Temporal Information (SI-TI) metrics seem to be more reliable than PSNR and Peak Error calculation. However no metric can be relied blindly to come up with a fair judgement. So, a standalone extremely reliable video metric, of course with reduced complexity, awaits further research. The Objective Difference Grade (ODG) and the Distortion Index (DI) are used to measure the quality of audio.

To demonstrate the different video variations for video AQoS *utilityfunction* we have implemented a number of AQoS instances, using different resolution scales, video bit-rates, and media colour types. Using the PSNR metric in the video utility function, we calculate the video quality. For doing that, we need both the original video stream and the adapted video segment frames. Experiments have shown that by tuning the scale, resolution, media colour and video bit-rate, the quality of the media is affected. For instance, we have noted that while resolution dropping does not always guarantee file size reduction, it may be used by clients with limited display capabilities in order to achieve a better screen quality. The greyscale factor decreases the file size and bandwidth for greyscale devices and based on content usage bit rate (Kbps) will also be optimally adjusted during adaptation. Even though in our experiments, adapted greyscale videos have greater quality (PSNR) than colour videos (figure 7), it doesn't mean that these videos are better with respect to quality. The reason being that we do not use the standard RGB Euclidean distance when we compare a colour and a greyscale video. Instead, we only calculate the difference of one colour since we assume that both videos use 8 bits per pixel. Furthermore, a user can consume a greyscale video at original resolution rather that seeing a colour video



	PSNR					
	Colour			Greyscale		
Video Bit Rate (Kbps) \ Scale	0	1	2	0	1	2
200	36	38.2	46	37	39	47
400	40	42.2	46.4	41	44	47.5
600	43	45.2	47	44	46	47.9
800	44	46.2	48	45.5	47.5	48.2

Fig. 7. PSNR

in a smaller resolution. As illustrated in figure 7, by tuning the video bit rate at lower resolutions (scale 2), video quality (PSNR) is negligible. Nevertheless, each adaptation option has different total value for each user. The user can specify their constraints and optimization objectives in a UCD file.

6 Conclusions and Future Work

The vision for MPEG-21 is to define an open multimedia framework that will enable transparent use of multimedia resources across a wide range of networks and devices. The big picture of how the suggested tools and mechanisms fit together is still not obvious, since the standard is still under development. In this paper we present our implementation of one of the most important semantic MPEG-21 tools, the AQoS. We have developed a flexible AQoS model for video adaptation and a set of algorithms for generating its parametric values. This paves the way for the design and implementation of an automatic video adaptation architecture that uses the AQoS, UCD and UED tools of MPEG-21 framework. The architecture will search AQoS switch parts for an optimal adaptation solution to enable an adaptation processor to process the description solution so as to provide the adapted video content.

References

1. ISO/IEC JTC 1/SC 29/WG 11: “MPEG-21 Digital Item Adaptation” N5845, 2004
2. Burnett, I., Walle, R.V., Hill, K., Bormans, J. and Pereira, F.: MPEG-21: Goals and Achievements. *IEEE Multimedia,* vol. 10, (2003), pp. 60-70.
3. Sikora, T.: Trends and Perspectives in Image and Video Coding. *Proceedings of the IEEE,* vol. 93, (2005), pp. 6-17.
4. Chang, S.F. and Vetro, A.: Video Adaptation: Concepts, Technologies, and Open Issues. *Proceedings of the IEEE,* vol. 93, (2005), pp. 148-158.
5. Kim, J., Wang, Y. and Chang, S.: Content–Adaptive Utility–Based Video Adaptation. Proceedings of IEEE Int'l Conference on Multimedia & Expo, (2003), pp. 281-284.
6. Mukherjee, D., Jae-Gon, E.K. Delfosse and Wang, Y.: Optimal Adaptation Decision-Taking for Terminal and Network Quality-of-Service. *IEEE TRANSACTIONS ON MULTIMEDIA,* vol. 7, (2005), pp. 454-462.
7. Timmerer, C. Hellwagner, H.: Interoperable Adaptive Multimedia Communication. *IEEE Multimedia,* vol. 12, (2005), pp. 74-79.
8. Panis, G. , Hutter, A., Heuer, J., Hellwagner, H., Kosch, H., Timmerer, C., Devillers, S. and Amielh, M.: Bitstream Syntax Description: A Tool for Multimedia Resource Adaptation within MPEG-21. *EURASIP Signal Processing: Image Communication Journal,* vol. 18, (2003) pp. 721-747.
9. FFMpeg Multimedia System retrieved from http://ffmpeg.sourceforge.net/ on 01/12/2005.
10. Richardson, I.E.G.: *H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia,* Chichester: John Wiley, 2003.

A Middleware Architecture Determining Application Context Using Shared Ontology*

Kugsang Jeong, Deokjai Choi, Soo Hyung Kim, and Gueesang Lee

Dept. of Computer Science, Chonnam National University, Gwangju, Republic of Korea
handeum@iat.chonnam.ac.kr, {dchoi,shkim,gslee}@chonnam.ac.kr

Abstract. Context awareness is a key part of ubiquitous computing. Recent middleware supporting it have the architecture to provide a context model to represent context information. The middleware recognizes contexts by using sensed and inferred information, applies them appropriately. This implies that the middleware should be able to determine all contexts of applications running. But since context-aware applications will be applied to wider areas and their number increases, it has become difficult for the middleware to determine all contexts needed for various applications. To overcome this, we propose architecture providing context definition by application using shared ontology. The middleware makes and maintains the shared ontology base in a ubiquitous computing environment. Applications write the context decision rule describing their own context and register it to the middleware. Then the middleware generates context objects to make a context decision according to the registered rule. If the current situation satisfies the rule, the context object notifies context information to a relevant application. Our application-defined context is middleware-independent so that it can make ubiquitous computing applications more capable.

1 Introduction

In the ubiquitous computing environment, context-aware applications must adapt to dynamically changing environments so that users can concentrate on their tasks without interference. Applications must be aware of their contexts in the environment [1]. There are two types of context-aware applications to adapt to current contexts: context-trigger and context-query. The context-trigger type defines contexts in which they operate, and perform specific behaviors whenever the current situation meets their context defined previously. The context-query type acquires the necessary context information during its runtime. In this paper, we have focused on context-triggered application.

The context-awareness in ubiquitous computing has been studied by both academics and companies for over 10 years. There are many examples of middleware that

* This research was supported by the Program for the Training of Graduate Students in Regional Innovation which was conducted by the Ministry of Commerce Industry and of the Korean Government Regional Innovation which was conducted by the Ministry of Commerce Industry and Energy of the Korean Government.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 128 – 137, 2006.

achieve context-awareness such as SOCAM [2], Aura [4], GAIA [5], and Context Toolkit [6]. Each middleware has its own context model to represent context information. Context-triggered applications should use the specific context information of middleware. Each application has interesting contexts of the target middleware, and writes logic using them so that it can be aware of and adapt itself to the changing contexts of the environment. The middleware in most recent research generate the context information for all applications by sensing or inferring information. Then they inform applications of the relevant context information whenever it changes or is requested. Under this architecture, the context-triggered applications should be developed after the context for applications have been defined and implemented by the middleware. So the application is tightly coupled with the pre-defined context of the middleware, otherwise known as middleware-dependent context.

Using the middleware-dependent context is not appropriate because it may be difficult, if not impossible, for the middleware to represent all contexts for ever increasing context-aware applications. The limitation of context scalability hinders context-aware application development. To make matters worse, if a new application is developed or an existing application should be changed in terms of using context, the middleware also might need to be modified.

To overcome problems from the middleware-dependent context like tightly coupled context decision-making and limited context scalability, we introduce the application context which means the context defined by an application. The application context is described as a set of ontology-based rules. Ontology is a common knowledge that represents all information of the ubiquitous computing environment [1, 2, 3]. It is sharable between the middleware and applications so that application developers can make rules to define application context without being dependent on a specific context model of middleware. The context-aware applications using application context is somewhat similar to context-aware services by pre-defined rules in SOCAM [2], but the difference is that applications using application context receive dynamically changing context information from middleware according to the results of rule evaluation while context-aware services use just pre-defined information in the rule. To adapt to dynamically changing contexts, applications should be able to receive dynamic information according to context changes.

The application should acquire interesting ontology information to make its own context decision by polling or asynchronous notifications from the middleware. The frequent communications between application and infrastructure may be a disadvantage, as is the computation overhead of an application to determine the context. As such, we have considered the delegation of making application context decisions. The application delegates the decision responsibility to the middleware by providing a rule set to evaluate. This rule set resides in the middleware and monitors the interesting contexts for applications by evaluating various pieces of simple context information. When the rule set is satisfied, the middleware notifies the interesting application.

To achieve our key ideas on the application context and the delegation of making context decisions, we propose a new middleware architecture making context decision according to a set of rules for application context. A context-triggered application has a context decision rule which defines its own context that an application wants to use. The context rule is written using a middleware-independent formal language based on the shared ontology. Our proposed middleware generates a context object per application to make a context decision according to the context decision rule given by an application. The context object provides the application with dynamically changing contexts whenever a current situation satisfies the rule.

In Section 2, we describe the application context and shared ontology. We present our proposed architecture in Section 3. Then we show an application example in Section 4. Lastly, in Section 5, we summarize the findings and provide a conclusion.

2 Application Context and Shared Ontology

The application context is the context which a context-triggered application is interested in. Context-triggered applications behave specific alls when a current situation in environments is found to be interesting. The Labscape application of one.world infrastructure defines application context as user's location changes so that it transfers experimental data to a computing device around a user whenever a user moves to a new location [5]. The application context is described by using a rule set based on shared ontology and determined by middleware. The rule and determination for the application context will be presented in the next section.

The shared ontology represents all information of a ubiquitous computing environment as sharable vocabularies. The middleware maintains the shared ontology on its environment and the application describes the rule for application context using instances of shared ontology. Recently, Semantic Space, SOCAM, and CoBrA have introduced the ontology to support knowledge sharing and context reasoning [1, 2, 3]. They also have developed their middleware using semantic web technologies like OWL [7]. The key idea here is the ontology-based context modeling that represents the vocabularies of sharable context information about the dynamically changing situation in a ubiquitous computing environment. Standardized ontology for

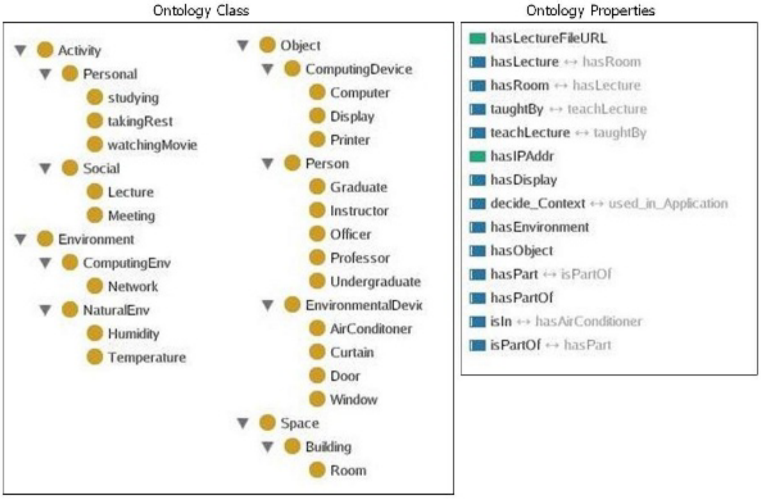


Fig. 1. Shared Ontology

ubiquitous computing is the first step and now being created [8], as there is currently no common ontology. We have used our own ontology-based context modeling for application context as a result.

Our ontology model has 4 classes: environment, object, space, and activity. The environment class represents computing and natural environmental information. The object class represents information of all objects in a ubiquitous computing environment, which includes computing devices like display devices and persons like students. The space class represents domain information like rooms and buildings. The activity class represents human activities in a specific domain like lectures and meetings. The relation between each class is described as properties. For example, an instructor having a lecture is expressed by a hasLecture property between instructor class and lecture class. Figure 1 shows our ontology model.

3 UTOPIA Architecture for the Application Context

In this section, we describe the architecture of middleware for supporting the application context, to be called UTOPIA (UbiquiTous cOmPutIng Architecture). UTOPIA allows each application to define middleware-independent contexts using shared ontology so that it can help developers create applications easily.

3.1 Overall Architecture

Our UTOPIA architecture consists of the following 5 components as shown Figure 2: Context Register, Context Object Generator, Shared Ontology Base, Ontology Event Provider, and Ontology Provider. The Context Register authenticates applications and accepts an application's context rules. The Context Object Generator generates an application-specific context object which evaluates the application's context rule. The Shared Ontology Base stores all context information in environments. The Event Provider signals context objects to re-evaluate a context rule whenever context information which context objects are interested in changes. The Ontology Provider wraps sensed or predefined information into context information, and provides the Shared Ontology Base with the information.

In terms of the steps of operation, an application first describes its application context using rule-based shared ontology and then registers the rule to the context register in the infrastructure. After authenticating an application, the context register hands it to the context object generator to generate an application context object. The context object generator makes the ontology event list to acquire dynamically changing ontology and registers it to ontology event provider. If the event rule is registered successfully, the application context object can subscribe asynchronously to the change of ontology instances which are used in its context rule. Whenever new information is received, the application context object evaluates whether changed situation meets the rule described in the context register. If the situation meets the rule, the application context object will notify an application of the context through context notification messages. The ontology provider stores sensed information from various and diverse sensors and pre-defined information into a shared ontology base. Simultaneously, the ontology provider should asynchronously inform the ontology event provider of the fact that ontology instances have changed in other words, that the situation of the environment is changed. Then the ontology event provider publishes those events to the related application context objects, so the application context object receiving ontology events can be aware of the situation change and can re-evaluate whether the changed current situation meets the rule.

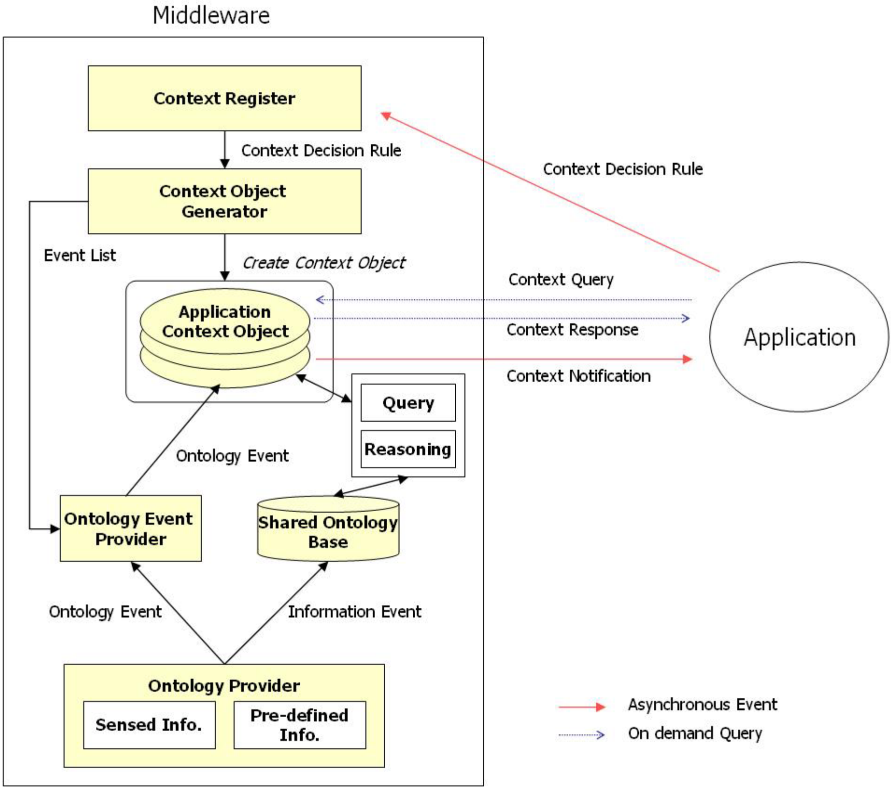


Fig. 2. UTOPIA Architecture for the Application Context

3.2 Context Decision Rule

The context decision rule is a set of rules to define the necessary application context. We use a human readable and horn-like form to be read easily. The context decision rule has the form: *antecedent => consequent.* The antecedent contains a sequence of conjunction of atoms written a_1 ^ … a_n. Atoms are classes representing sharable ontology between middleware and applications in ubiquitous computing environments. Variables in the rule are prefixed with a question mark, e.g., ?*x* for variable *x*. The consequent is an application context statement that will be notified to an application when the result of antecedent evaluation is true.

Table 1 shows an example of the context decision rule of the ULecture application described later. This application's context is that lecture starts when an Instructor enters a lecture room. The isIn (?*x*, ?*y*) atom is a property between a person and a space which represents a person *x* in a space *y*. The Instructor (?*x*) atom refers to person *x*'s identity as an instructor, and the LecutureRoom (?*y*) atom means that the space *y* person *x* enters is a lecture room. If the antecedent of the context rule below is true, consequent, "Lecture start," is notified to the ULecture application with names of instructors and rooms

Table 1. Context Decision Rule Example

horn-like context decision rule: isIn(?*x*, ?*y*) ^ Instructor(?*x*) ^ LectureRoom(?*y*) => Lecture_start
Semantics: When someone enters a specific space, If (someone is an instructor and space is a lecture room) Then a lecture will start

3.3 Context Register

The context register waits for a request from an application and authenticates the application with the ID and password. The first step of an application is to contact the context register of middleware. To do this, we have implemented a simple context register discovery protocol similar to DHCP. After finding the location of the context register, with an IP address and port number, an application starts the authentication process. If the authentication is successful, an application can register its context decision rule to the context register. After verifying the registered rule's syntax, the context register then hands the context rule and application's location to the context object generator.

3.4 Context Object Generator and Event List

The context object generator has two roles: the first is to generate the application context object, and the second is to make the event list for the ontology event provider to be able to notify the dynamic change of the environment to the application context object.

The event list for each application consists of the ID indicating an application context object and ontology properties described in the context decision rule. Whenever environments change, instances of ontology property also change. To recognize the change of environments, the application context object acquires events for property instances' change according to event lists. The ontology event provider monitors continuously the instances of ontology properties described in the event list.

3.5 Application Context Object

The application context object makes a context decision and responds to a context query from context-triggered applications. All applications should have one context object which is responsible for their context decisions and queries. Each context object stores a set of rules describing application context, and listens to ontology instances' changes and query requests. The context object is generated when an application registers rules to middleware and terminated when the application is over. The functions of application context object are shown in Figure 3.

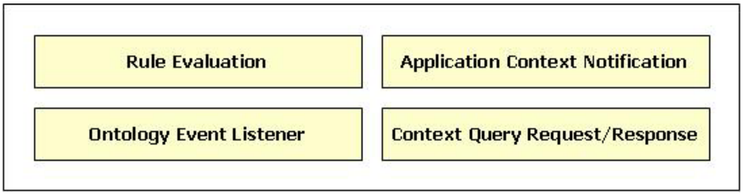

Rule Evaluation
Application Context Notification
Ontology Event Listener
Context Query Request/Response

Fig. 3. Functional Architecture of Application Context Object

The context object evaluates the rule set whenever an environment changes and makes a decision about whether a relevant application is interested in current context or not. The context object listens to events from the ontology event provider. When receiving events, the context object evaluates conditions in the rule's antecedent part. The conditions are converted to query statements to query values of ontology instances by a query engine. If the result of a query is false, the context object determines that the current context is not the relevant application's context of interest. If the result has one more value, the context object determines that current context is the application context and notifies the application context including result values to the relevant application. The application can adapt itself to dynamically changing environments by behaving according to application context notified by the context object.

The context object accepts the request for a context query from a relevant application. After receiving a query statement, the context object performs immediately the query through the query engine and transfers results to the application. The application then makes and delivers a query statement to the context object. Figure 3 shows the function

We have implemented the context object as a java object that evaluates the context decision rule. The antecedent of the context decision rule is actually converted into a query statement which conforms to Bossam's syntax in the Context Object. Bossam is an OWL inference and query engine [9]. To make a context decision, the context object performs the converted query using the Bossam engine. For example, query statement to the antecedent of the rule in Table 1 is "query q is isIn(?*x*, ?*y*) and Instructor(?*x*) and Room(?*y*);." The response of the query execution would be *false* or instance list of variable *x* and *y* meaning that the antecedent of rule is true. If the response is not false, the Context Object informs the corresponding application of the context including the query results. The query results of Bossam are a little verbose so we make it simple as a list of "variable=value" like "x=student_0 x=student_1." The statement for context query is the same as shown in the above example. For example, a query for a list of students at room 410 can be described as "query q is isIn(?*x*, room_410) and Students(?*x*);".

3.6 Ontology Provider

The ontology provider stores sensed or pre-defined information into the shared ontology base. The sensed information is obtained from various and diverse sensors deployed in a ubiquitous computing environment. The pre-defined information is obtained through database or files stored manually. The semantic web technology for ontology does support knowledge sharing, query, and reasoning well, but do not support asynchronous events [7]. As such, we have added another role into the ontology provider. When storing information into the shared ontology base, the ontology provider generates the ontology event and notifies the ontology event provider if the instances of ontology are changed.

3.7 Shared Ontology Base

The shared ontology base provides consistent context knowledge storage. As described, we have a context model having 4 classes. UTOPIA maintains a file as the shared ontology base. The shared ontology stores instances' values into a file which can be accessed by all context objects that evaluate their rules. We used the Protégé ontology editor to build context modeling and create an OWL file, and update instances' values by using API of Jena [10] and Protégé OWL plugin [11].

3.8 Ontology Event Provider

The ontology event provider lets the application context object know of environmental changes. The ontology provider receives ontology events issued from the ontology provider and also maintains event lists from the context object generator. Ontology events have instances' values changed. The event list presents ontology instances which the application context object monitors. According to the event list, the ontology event provider determines which application context object it should redeliver events to when ontology events are received.

4 ULecture Application Example

In this section, we show a context-triggered application, ULecture, using middleware-independent application context. It proves that our proposed application context makes application development easier and simpler.

ULecture is an application for lecture room in a ubiquitous computing environment. ULecture defines the application context as the rule described in Table 1. The rule says that if an instructor enters a lecture room, then the application context is notified to the ULecture application. After simple authentication processing, the ULecture registers its context decision rule to the context register. If there are no problems in rule verification, the context register delivers the rule to the context object generator to generate a context object for the ULecture. The context object generator also makes event list to register it into the ontology event provider. When a sensor at the door senses someone entering a room, the ontology provider adds property instances, that is isIn (Mr. J, Room_401) in shared ontology base, and at the same time notifies ontology events to the ontology event provider. Because isIn property of shared ontology changes, the ontology event provider notifies this change to the context object for the ULecture. Then the context object re-evaluates the context decision rule. If Mr. J is an instructor and Room 401 is a lecture room, then the result of rule evaluation is “Instrucutor=Mr. J LectureRoom=401.” The context object then notifies the application context, “Lecture starts” with the result of rule evaluation to the ULecture. After receiving an application context notification, the ULecture performs actions to provide services like downloading a lecture material and projecting it on a beam projector. Figure 4 shows the steps described above. Under UTOPIA architecture, application developers simply know how to describe their context decision rule based on shared ontology to receive application context asynchronously.

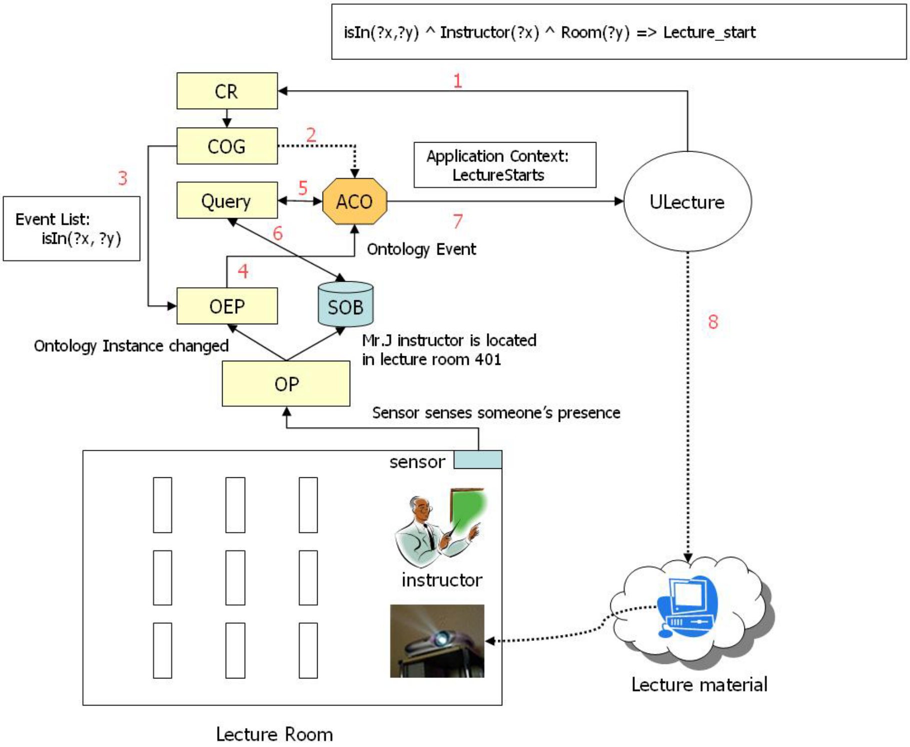


Fig. 4. ULecture Application Example

5 Conclusion

We proposed the UTOPIA architecture for supporting application context. In our architecture, applications define their context of interest as a rule based on shared ontology and register it to delegate its evaluation to UTOPIA. To define application context, we introduce the context decision rule to be written using horn-like rule based on shared ontology.

We expect that UTOPIA will help to overcome problems from middleware-dependent context like limited context scalability and tightly coupled context decision-making. An application developer who wants to develop new context-aware applications can define simply an application context rule based on the classes and properties of the shared ontology for a new context. As a result, UTOPIA provides an easy way to develop context-aware application without understanding of internal structure of the middleware.

References

1. Wang, X., Jin, S.D., Chin, C.Y., Sanka, R.H., Zhang, D.: Semantic Space: An Infrastructure for Smart Spaces. PERVASIVE computing (2004)
2. Gu, T., Pung H.K., Zhang, D.: A Service-Oriented Middleware for Building Context-Aware Services. Journal of Network and Computer Applications (JNCA) Vol. 28. (2005) 1-18
3. Chen, H.: An Intelligent Broker Architecture for Pervasive Context-Aware Systems. PhD Thesis, (2004)
4. Soursa, J.J., Garlan, D.: Aura: an Architectural Framework for User Mobility in Ubiquitous Computing Environments. the 3rd Working IEEE/IFIP Conference on Software Architecture (2002)
5. Roman, M., Hess, C., Cerqueria R., Ranganathan, A., Campbell, R., Nahrsted, K.: Gaia: A Middleware Platform for Active Spaces. IEEE Pervasive Computing (2002) 74-83
6. Dey, A.K.: Providing Architectural Support for Building Context-Aware Applications. PhD Thesis, (2000)
7. McGuiness, D.L., Hanmerlen, F.: OWL Web Ontology Language Overview. W3C Recommendation 2004.
8. Chen, H., Perich, F., Finin, T., Joshi, A.: SOUPA: Standard Ontology for Ubiquitous and Pervasive Applications, International Conference on Mobile and Ubiquitous Systems: Networking and Services (2004)
9. Jang, M., Sohn, J.: Bossam: an extended rule engine for the web, Proceedings of RuleML, LNCS Vol. 3323 (2004)
10. Carroll, J.J., Dickinson, I., Dollin, C.: Jena: Implementing the Semantic Web Recommendations, tech. report HPL-2003-146 (2003)
11. Knublauch, H., Fergerson, R., Noy, N., Musen, M.: Protégé OWL Plugin: An Open Development Environment for Semantic Web Applications, Third International Semantic Web Conference (2004)

Context-Aware Regulation of Context-Aware Mobile Services in Pervasive Computing Environments

Evi Syukur[1] and Seng Wai Loke[2]

[1] Caufield Institute of Technology, Monash University, Melbourne, Australia
Evi.Syukur@csse.monash.edu.au
[2] Department of Computer Science and Computer Engineering, La Trobe University, Melbourne, Australia
S.Loke@Latrobe.edu.au

Abstract. This paper discusses concepts, the design and prototype implementation of a context-aware policy system that governs mobile services visibility and execution in pervasive computing environments. We view a pervasive environment as a collection of mobile users, mobile services, contexts, policies and roles. Applicable policies are selected depending on the context of users i.e., location, activity and the user's role, and policies determine what services one can see and access in different contexts. Our approach provides a generic context-based regulated mobile services execution model, applicable to different pervasive domains (e.g., a Campus domain).

1 Introduction and Related Work

As a pervasive system often consists of a number of entities (e.g., many users, rooms, etc), balancing control given to one user (entity) by considering other entities' activities or situations is important. This becomes paramount in pervasive environments, as users tend to be on the move (i.e., from one geographical location to another) and are allowed to access services or perform actions on the service at any time and any place that the user visits. The flexibility and convenience given to a user to access and perform actions on the service need to comply with a current and relevant global policy specified by the system (or a space).

An approach to enforce an access control for mobile services is by having a sensor system (e.g., a location sensing system). For example, if the owner of the room (e.g., user A) is not in her room, the system automatically allows visitors to start any service at room A, and when, the system detects user A is approaching the room, the system then prohibits visitors from starting any service or warns visitors that the running service need to be stopped (as the owner requires a quite working environment to focus on the work). This approach is simple to implement and maintain, as no policy rule is required. It proactively starts/stops services depending on the owner's current proximity (that is retrieved from a location sensor system). However, it does not support complex situations such as: (a) what happens if I only allow visitors with the same level of role to start the service, (b) although I am not in the room, I do not allow them to start any service unless, they are my students. The above more complex

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 138 – 147, 2006.

situations can be realized with the use of policy or rule that specifies what activities or actions that a space allows, or prohibits visitors from doing so in particular contexts.

Integrating a context-aware system with a policy subsystem offers several advantages such as the system or space having control over users' (e.g., visitors') behaviours in accessing services in particular contexts and hence, can limit conflicts that may occur between users who are trying to access the same service (e.g., a music service) on the same shared resource device with different actions (e.g., one wants to stop and another wants to start the music service). It also restricts users from performing prohibited actions at specific contexts.

Many existing policy projects focus on security based notions [1,2,3], where is defined a set of activities that an authorised user can do (permission), can not perform (prohibition) and must perform (obligation) within an organization depending on the position or level that the user has, often called role based access control [4]. Our initial policy work has been discussed in [8], where we illustrate the usefulness of having policy to govern the execution of mobile Media player service (that is based on the user's contexts such as: location, identity, day and time). There are a few limitations in our initial policy design, where (a) the policy is not reusable across spaces, (b) the policy only controls the user's execution (not the service visibility), (c) the policy is limited to a user's preferences, we have not taken into account the environment or space policy, (d) it has not taken into account the logic of norms (i.e., permissions, obligations and prohibitions given to users in particular situations), (e) the policy only targets certain users (e.g., does not have a concept of role), and does not have a space concept (a space can be a public or private space).

This paper greatly extends our initial idea, where not only service is aware of the user's context, but, also the policy. In this version, policy can be used to govern and control the behaviours of entities in accessing mobile services (e.g., service visibility and actions allowed) in specific contexts. We are not only controlling access to devices if accessed via services (e.g., whether a user is permitted to use the power point presentation program on machine A or whether a user can modify the log file on computer A), but also regulating the behaviours of services (not only access but also execution). Also, we note that such regulation (consists of permissions, obligations and prohibitions) itself depends on the context. We also incorporate much richer contexts (than in our previous work) such as a user's identity, current location, day, time and activity. In addition, our policy design also supports different levels of users (based on their roles); is reusable in different spaces; supports much richer target entities (i.e., not only a user but also a system and space).

In our work, a mobile service is a software tool that provides assistance to users to complete their daily tasks. For example, a Mobile pocket pad service [6], a Mobile teleporting service [7], an Auto task service and a Mobile media player service [8]. Many existing policy work in pervasive environments focus on different types of context such as context of an agent [13], and security [5, 14], rather than context of the user, and does not deal with mobile services. Two closely related policy projects that use policy and services are Active Space [12] and Task Computing project [9] that uses Rei Policy Language [10]. However, the purpose of policy here is different from our work. They employ policy to ensure that users only use and access resources (i.e., devices) in authorized ways within a space, but we also use policy to impose required service behaviours. In addition, the policy in our work determines what services one can see and access in different contexts. The policy itself is context-aware, in which, different policies will be applied (enforced) in different situations (contexts) in pervasive environments. For example, our system only prohibits students to access lecture note service during exam time, other times (before and after the exam), the user is able to access all services as per normal.

This paper discusses the idea, concept, and prototype implementation of regulated context-aware mobile services execution in the context of the implemented Mobile Hanging Services (MHS) framework. The rest of this paper is organised as follows. In section 2, we discuss main requirements and issues for building context based policy systems. In section 3, we provide our conceptual model. In section 4, we discuss an implementation of MHS. In section 5, we discuss our performance result. In section 6, we conclude the article and present future work.

2 Background on Policy

There are several main roles of a policy (access control) in pervasive environments:

(a) to define the visibility of services in particular context i.e., two users with different roles may see different services available in the same context.
(b) to constrain the behaviours of foreign agents or visitors accessing services in the user's room (i.e., to protect a user's privacy and give the owner of the room the ability to control the activities of visitors in his/her room).
(c) to help users to perform a task automatically within a certain situation (i.e., a policy rule can say "automatically start the music (service) at 12:30PM at room A, playing The First Noel").
(d) to control the behaviours of entities in executing a service, especially a shared resource service, in which there are multiple users try to control the execution of a service with different interests (i.e., one user wants to start music A, but, another user wants to start music B on the same embedded music device). These differences lead to a conflict and require a resolution. The conflict can be avoided and limited by assigning different level of control to different users (e.g., a user with higher level role can see any service, but a user with lower level role can only see certain service.

Several issues specifically in designing policy in pervasive environments are:

a. Users are always on the move and more context information need to be collected before applying the right access control.
b. Many conflicts may occur in pervasive environments, due to more contexts used, mobility of users and services. The conflicts need to be detected and resolved in an appropriate manner seamlessly with minimal or without user intervention [11].

3 Conceptual Overview

We model our policy system where the policy enforcement depends on the user's current contexts (e.g., location, activity, day, and time). For example, when a user enters a campus domain such as the entrance of the campus, the campus policy is applied. When the user enters Blackwood building, the building policy is applied and when the user enters a lecture hall, then the lecture hall policy is applied. Note that a lecture hall might also have different policies at different times and occasions.

We now discuss seven main elements of our system:

a. Policy objects. Policy objects are based on the logic of norms for representing the concept of rights, obligations and prohibitions in our system. Right (R) refers to a permission (positive authorization) that is given to an entity to execute a specified action on the service in the particular context. Obligation (O) is a duty that an entity must perform in a given context. Prohibition (P) is a negative authorization that does not allow an entity to perform the action as requested.

b. Mobile services. Mobile service refers to a particular service that the user wants to execute (e.g., a mobile pocket pad service, mobile VNC service, etc.).

c. Actions. Currently, there are four actions, which are commonly used in our system. These are start, stop, pause, and resume. We generalize and group the action according to its purpose. For example, in Media Player Service, we have *"a play button"* to start the music, but, in Mobile VNC service, we have *"a start button"* to begin teleporting. These "start" and "play" actions have the same meaning and purpose (to start the service) though it is called differently from each service.

d. Contexts (domain constraints). Contexts are conditions that must be met before a list of services can be displayed on the mobile device or before the user's request to perform an action is approved. Currently, contexts in our work consist of a user's identity, location, activity, day and time.

e. Role. Role is associated with a level of privileges that determines the actions that a user can perform and the visibility of services in particular contexts. Depending on the role that the entity has, s/he may have different privileges in executing the service. For example, a user with higher role can do more things and have more services available compared to the user with lower role. In our system, we classify users into three different roles: a super entity (e.g., a head of school), power entity (e.g., staffs) and general entity (e.g., undergraduate and postgraduate students).

f. Space Policy. One space policy (e.g., room A's policy) in a system may be different from another space policy (e.g., room B's policy). This all depends on the purpose and activity that is currently running in the space. Two types of space in our system are:

(i) Public space such as a tea room, a seminar room, an exam room, etc. The policy for a public space is written by a developer (system entity) and is bound to all users who enter a public space. The system entity controls users' behaviours in accessing or executing services in the public space.

(ii) Private space such. The private space policy is written by the owner of the room. All visitors (including colleagues) who visit a private space (e.g., Alice's room) are bound to rules specified by the owner of the space and so, they are not allowed to execute any service or perform any action if not permitted by the owner of the space. This is useful, as in some situations an owner would like to be in control of all visitors' behaviours in her space (e.g., during study hours, no services can be played).

g. Target entity. A user is an active entity that is always on the move (able to move from one geographical place to another). By default, the space (i.e., public space or private space) imposes on the user certain rights (denoted by spRu), obligations (spOu) and prohibitions (spPu) for each physical location in the system based on the users' role and current activity. On top of this default space policy, users can also specify a set of obligations to the system (denoted by uOsp), created via a user policy application that we have. Imposing a set of obligations on the system is a means for the user to have the system perform tasks automatically on behalf of the user at a certain location, day and time. This then helps to reduce the user's task especially if there is regularity in the user's activities [5]. Our system also checks for the conformance of uOsp against the permission that the space gives to the user in that specific location. This is to ensure that the uOsp is still within the scope of the user's permissions.

Policy Language Design

In designing a policy language, it is important to balance the convenience and compliance aspects, where a system has control over users' actions or activities, but does not overly restrict or control users' behaviours. This is possible by specifying rule per activity, in which only at certain occasions, the space will be in control. Ideally, the end-user would still be able to access services as per normal in all public places and circumstances, and only in some situations (e.g., during exam or meeting time), the space takes full or partial control over the service from users (e.g., allowing users to perform certain actions on the service or prohibiting users from performing any action on the service) as illustrated in Figure 1 below.

In addition, our policy design also takes into account the reusability aspect, in which the policy is stored on the server side and can be shared with other spaces in the system. This is possible, as in creating rule per activity, we do not explicitly specify the context information (e.g., space/location as well as the exact date and time of when and where the activity occurs). Instead, we store this context and activity mapping in an external file (see Figure 2 below). The mapping here works like a booking system, where it stores the user's schedule (in this case the owner of the space's or the public space's activities). The system then refers to this location_activity document to have an idea of the activity running in the space. After that, it retrieves the relevant rule that matches this activity. It then enforces the rule to all users who visit the space when the contexts elapse.

This activity information can also be retrieved from sensing devices installed in the environment (e.g., using smart camera that could detect the user‘s activities and movements). We will continue to integrate this smart sensing device into our contextual system in the future. The followings are a sample of a space policy document in an XML language based on activities that may occur in a space (as illustrated in Figure 1 below). We also give a sample of how the mapping between location, activity, day and time in our system (see Figure 2 below). The mapping and policy are created by the developer or owner of the space. The mapping is done per space (to customize the activities that may occur in the space), but, a generic policy rule can be shared. The advantage of separating the rule and context details is the rule does not have to be changed when the activity and contexts change, only the mapping needs to be updated when there is a new event or modification of an existing event. The rule can also be re-used by other spaces which have the same activity. This is possible as we have a consistent naming of activity throughout all spaces.

In a case, where the activity at certain day/time is not specified in the mapping document (e.g., between 12-1PM and after 2PM as shown in Figure 2 below), the system then looks for activity name="any" in the policy rule. During this time (activity="any"), all visitors are given flexibility to access any service and perform any action. This then balances the convenience and compliance aspects in our system, where the space is only in control at some situations (activities), and the rest users could still access services as per normal. In addition, "any" on *service allowed* means any service as described in the user's preferences for that particular contexts, "any" on action means any action that a service supports (e.g., a media player service has start, stop, pause and resume actions). "None" simply means no services will be visible or no actions are allowed at certain activity.

```
 <Rule>
   <Activity name="Meeting">
       <Has policyObject="Right" by="System" on="General_User">
            <Service allowed="Mobile Pocket Pad Service">
               <Action allowed="Any"/>
             </Service>
       </Has>

       <Has policyObject="Obligation" by="System" on="General_User">
             <Service obligated="any">
               <Action obligated="stop"/>
             </Service>
       </Has>

       <Has policyObject="Prohibition" by="System" on="General_User">
             <Service prohibited="any">
                <Action prohibited="any"/>
             </Service>
       </Has>
   </Activity>

   <Activity name="Any">
       <Has policyObject="Right" by="System" on="General_User">
            <Service allowed="any">
               <Action allowed="any"/>
             </Service>
       </Has>

       <Has policyObject="Obligation" by="System" on="General_User">
             <Service obligated="none">
               <Action obligated="none"/>
             </Service>
       </Has>

       <Has policyObject="Prohibition" by="System" on="General_User">
             <Service prohibited="none">
                <Action prohibited="none"/>
             </Service>
       </Has>
   </Activity>
</Rule>
```

Fig. 1. A sample policy document

```
<Location_Activity for="RoomB530" createdBy="Alice">
    <Activity_Details day="Monday" time="9-12PM">
          <Activity name="Meeting"/>
    </Activity_Details>
    <Activity_Details day="Monday" time="1-2PM">
          <Activity name="Out to lunch"/>
    </Activity_Details>
</Location_Activity>
```

Fig. 2. A sample location_activity mapping document

4 Prototype Implementation

The system consists of users with handheld devices, a desktop machine for executing a shared resource service, a Web service that determines the location of a user, a set of location-based services and policy functionalities that are published via the system. Our policy implementation is developed on top of our previous mobile services prototype [6]. The policy software components only get called when the service interface has been displayed and the mobile user is requesting to execute a certain action on the service i.e., by clicking on the start button on the media player service interface on the mobile device[1]. Our policy implementation is modular, interoperable and scalable. We separate the policy tasks according to its functionality i.e., we have a separate web service method for policy interpreter, conflict detection, resolution and manager. Hence, we only need to update a single component (i.e., the context collection component) if there is a new context added in the future. In addition, we also separate the policy implementation from the services (or their mobile code) implementations and store the policy specification on the server side. This allows our system to easily add additional services in the future and we may need to have only one policy document for all services or applications that we have in the system. Moreover, as we create each of our software components as web services, this makes our software functionalities accessible in disparate platforms and languages. Figure 3 below describes in detail on how our policy mechanism works.

The steps in Figure 3 are:

1. Request an action (e.g., start) on a service

Once the service interface is displayed on a mobile device, a mobile user can request to start a music on a media player service by clicking on the "start button". When there is a request from the user, the mobile client query manager then passes this query on to a policy manager (i.e., to decide whether or not the user is permitted to start the service with a particular song name).

2. Call the policy interpreter

There are a few steps needed to be performed by a policy manager in order to answer the user's query such as calling the policy interpreter to collect information regarding the user's current context and the relevant policy documents.

[1] One could also think, a mobile user interacts with music device that is embedded on the wall via voice command. A policy mechanism gets called once, there is a request from a user to perform certain action on the music service.

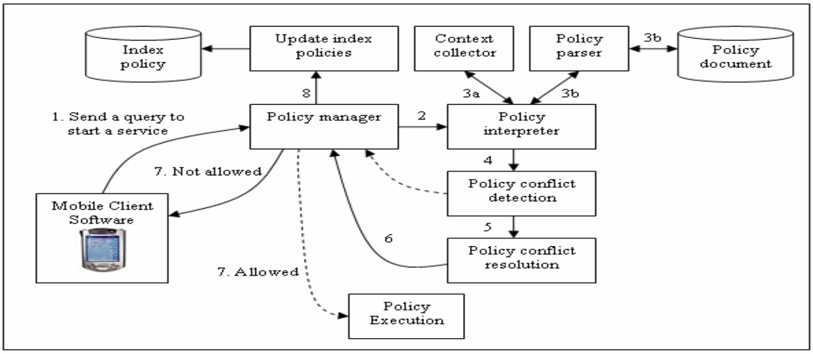


Fig. 3. MHS Policy implementation

3a. Retrieve the context information

First, the policy interpreter contacts the context collector to retrieve a user's current context information such as location by calling the Ekahau location server, activity (e.g., having an exam, giving a presentation) by calling the booking system Web service and other contexts used in the system (e.g., current day, time and a user's identity). The booking system keeps track of the user's location-activity information (e.g., what and when an activity occurs). The updated current day and time are obtained by checking the current system day and time. The user's identity is retrieved from the login form, once the user logs on to the system.

3b. Retrieve the entities' policy documents

Once, the system manager receives all context information from context collectors, this information is then passed on to a policy manager to find the relevant policy information based on the user's current context. The policy manager then interprets and caches the relevant policy decision result on the server side for future re-use (e.g., for other users who have the same role and context). The policy decision specifies list of services that a user can access as well as list of actions that can be performed.

4. Run time policy decision checking

Our system employs a group checking mechanism, in which, policy checking does not need to be done on each action requested, but is done on a group of actions. Checking per action is redundant and may not be necessary due to: (1) users may click on the same action more than once, and (2) for some services (e.g., mobile VNC service, allowing users to start the service, would also require a permission to stop the service, and on presentation service, allowing users to navigate the slides forward, would also require a permission to navigate backward). As a result, the user would be able to see the executed action immediately (with minimum delay). In policy checking, a requested action is only allowed if there is permission given by the space. If no permission is given (e.g., permission does not specify the requested action), this simply means the requested action is not allowed and hence, the policy manager does not need to check on the prohibition.

5. Call the conflict resolution

If there is any conflict detected in steps i, ii, iii, or iv above, this conflict detection result will be passed onto the conflict resolution to be resolved. Our conflict resolution resolves all conflicts detected and caches this resolution result for future re-use.

6 & 7. Send the result to the policy manager and back to the client manager or call a policy execution.

If no conflict is detected, the conflict detection module then sends a message to the policy manager (i.e., allowing user A to execute the specified action as no conflict has been detected, and so, no resolution is required). This then calls the policy execution service to perform a specified action on the service (i.e., start playing “First Noel” at any nearby desktop machine at room A). However, if there is a conflict, and the result stated that the user is not allowed to perform the specified action, the policy manager then sends back a message to the mobile client manager.

8. Update the index policy

The index policy database only gets updated when there is a permission to perform an action, and so, the state of service is changed accordingly (i.e., changing the state of the service from idle to “running” or from “running” to “not running”).

Our policy performance evaluation has been discussed in detail in [11].

5 Conclusions and Future Work

We have presented a context-aware policy model for context-aware mobile services access and execution, which can take into account the preferences of different users and spaces (or their owners/caretakers). Not only are services offered to users context-dependent, but the policy for regulating the services are also context-dependent (e.g., different policies for different spaces, and perhaps even for different times in the same space). Future work involves iterating over the conceptual design, as well as investigating approaches to improve the efficiency and the effectiveness of the prototype implementation: (a) security in transferring the policy file, (b) integrity of a cached policy file on the user’s mobile device.

References

[1] Lymberopoulos, L., Lupu, E. and Sloman, M., “PONDER Policy Implementation and Validation in a CIM and Differentiated Services Framework”, Proc. of the 9th NOMS 2004, May 2004, Korea.

[2] Godik, S., and Moses, T., “OASIS eXtensible Access Control Markup Language (XACML)”, OASIS Committee Specification cs-xacml-specification-1.0, November 2002.

[3] Uszok, A., Bradshaw, J., Hayes, P., Jeffers, R., Johnson, M., Kulkarni, S., Breedy, M., Lott, J., and Bunch, L.,”DAML reality check: A case study of KAoS domain and policy services”, ISWC 03, Sanibel Island, Florida, 2003.

[4] Sandhu, R.S., “Role-based Access Control”. In Zerkowitz, M., ed.: Advances in Computers. Volume 48. Academic Press (1998).

[5] Muhtadi, J., Ranganathan, A., Campbell, R. and Mickunas, D.,“Cerberus: A Context-Aware Security Schema for Smart Spaces”, *Proc. of PerCom 2003*, Dallas-Fort Worth, Texas, March 23-26, 2003

[6] Syukur, E., Cooney, D., Loke, S.W. and Stanski, P., “Hanging Services: An Investigation of Context-Sensitivity and Mobile Code for Localised Services”, Proc. of the IEEE MDM Conference, USA, Jan 2004, pp.62-73.

[7] Syukur, E., Loke, S.W. and Stanski, P., “The MHS Framework for Context Aware Applications: An Experience Report on Context Aware VNC”. Technical Report no:151/2004, Monash University, Australia.

[8] Syukur, E., Loke, S.W. and Stanski, P., “A Policy based framework for Context Aware Ubiquitous Services”, Proc. of the EUC Conference, Aizu-Wakamatsu, Japan, LNCS, vol. 3207, Springer-Verlag, pp.346-355, 2004.

[9] Masuoka, R., Chopra, M., Labrou, Y., Song, Z., Chen, W.I., Kagal, L. and Finin, T., “Policy-based Access Control for Task Computing Using Rei”, Proc. of Policy Management for Web Workshop, Chiba, Japan.

[10] Kagal, L., “Rei[1]: A Policy Language for the Me-Centric Project”, HP Laboratories Palo Alto, HPL-2002-270, 2002.

[11] Syukur, E., Loke, S.W., and Stanski, P., ”Methods for Policy Conflict Detection and Resolution in Pervasive Computing Environments”, Proc. of Policy Management for Web Workshop, Chiba, Japan.

[12] Sampemane, G., Naldurg, P. and Campbell, R.H., “Access Control for Active Spaces”, Proc. of the 18th *ACSAC*, 2002

[13] Bellavista, P., “Mobile Agent Models and Technologies for Distributed Coordinated Applications in Global Systems”, PhD Thesis, University of Bologna.

[14] Becker, M.Y., and Sewell, P.,“Cassandra: Flexible Trust Management, Applied to Electronic Health Records”, In *Proc. of the 17th IEEE Computer Security Foundations Workshop*, 139–154, 2004.

Designing and Implementing Physical Hypermedia Applications

Cecilia Challiol[1], Gustavo Rossi [1,3,*], Silvia Gordillo[1,2], and Valeria De Cristófolo[1]

[1] LIFIA, Facultad de Informática, UNLP, La Plata, Argentina
{ceciliac, gustavo, gordillo, valeriac}@sol.info.unlp.edu.ar
http://www-lifia.info.unlp.edu.ar
[2] Also CICPBA
[3] Also CONICET

Abstract. In this paper we present a design approach and a software framework for building physical hypermedia applications, i.e. those mobile (Web) applications in which physical and digital objects are related and explored using the hypermedia paradigm. We show how we extended the popular MVC metaphor by incorporating the concept of located object, and we describe a framework implementation using Jakarta Struts. We first review the state of the art of this kind of software systems, stressing the need of a systematic design and implementation approach; we briefly present a light extension to the OOHDM design approach, incorporating physical objects and "walkable" links. We next present a Web application framework for deploying physical hypermedia software and show an example of use. We compare our approach with others in this field and finally we discuss some further work we are pursuing.

1 Introduction

A physical hypermedia (PH) application is a kind of ubiquitous software in which the mobile user can explore real world objects using the hypermedia paradigm. In these software systems, physical objects are augmented with digital information in such a way that when the user is in the vicinity of an object, he can access the additional information. Besides, physical objects can be seen as nodes in a hypermedia network; the user can follow a link to navigate to other related objects, either virtually, e.g. when the links are implemented using a Web browser, or physically by moving to the target object.

A simple example is a mobile tourist guide. When the user is in front of a monument he can read information about the monument in his mobile device (e.g. in a Web page); he can also explore the digital hyperspace by navigating to other related documents. Some links, however, may point him to other tourist spots in the same city. Instead of navigating in the usual digital way, he has to "walk" the link [9]. The software system may react to his intention to navigate by providing him a map showing the best way to access the target place. Notice that this application behavior, while somewhat similar to existing families of location-based services, is completely based

[*] This work has been partially funded by Project PICT 2003, Nro 13623, SeCyT.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 148 – 157, 2006.

on the well-known ideas of hypermedia navigation that became popular with the Web. It is not surprising then that the physical hypermedia paradigm has been considered to be a good vehicle to integrate the Web and the world [8] and as a tool to improve collaboration in a social setting [3].

We have been working on different aspects of the PH applications' life cycle. In [6] we presented a modeling and design approach that allows a high-level specification of the intended functionality of a PH application. In [5] we analyzed a more complex engineering aspect of this kind of software: how to clearly decouple the most critical concerns that designers face when building PH software. We defined the concept of concern-driven navigation to support the user while exploring different application themes and to clearly separate digital from physical navigation.

In this paper we present a software framework that allows seamless implementation of PH applications. This framework, which implements an extension of the popular MVC metaphor, has been built on top of the well-known Jakarta Struts [16] Java infrastructure and therefore can be easily used by Web application designers.

The main contributions of this paper are the following:

- We present a reusable software substrate that supports the main abstractions in the PH paradigm,
- We show how to integrate this framework with a modular design approach thus covering the full PH software life cycle.
- By describing the implementation of a simple application we introduce a set of good design practices that help the implementer to cope with the difficulties that arise while building this kind of ubiquitous web software

The rest of the paper is organized as follows: In Section 2 we describe the requirements of an implementation framework for PH and present some background concepts needed in the rest of the paper. In Section 3 we present our extension to the MVC metaphor and describe our framework implementation. In Section 4 we present an example of use of the framework. In Section 5 we compare our work with other similar approaches and in Section 6 we present some concluding remarks and further work on this area.

2 Requirements and Background

Researchers have emphasized the feasibility of the PH paradigm by building software infrastructures that support the ideas underlying PH [7, 8, 13]. However, we think that for PH applications to become mainstream we need to provide tools to build them as a particular class of mobile Web software, i.e. we need to specialize existing Web development architectures and frameworks in such a way that they support the main concepts behind PH. We also need to provide a conceptual framework to reason on this kind of software and a comprehensive bridge among modeling concepts and the implementation tools. Particularly, a development framework must:

- simplify the development of this kind of software, providing reusable classes and semi-complete application structures together with "hot-spots" in which developers can add the specific aspects of their own applications,

- allow a clear separation between application objects and the lower level aspects needed to indicate their physical position and to check whether a user is in front of an object,
- support different navigation strategies, such as digital (as in the Web) or physical, allowing the designer to easily implement both of them,
- provide ways to maintain basic contextual information, e.g. when the user navigates digitally, keep the physical links corresponding to the current location visible.
- Additionally, supporting different mobile devices is a must; finally, applications built using the framework should also support conventional access, e.g. from a desktop browser. In the following sub-sections we briefly describe some background concepts that we will use throughout the paper, namely the philosophy underlying our design approach, and the basic concepts behind the MVC metaphor.

2.1 Design Issues for Physical Hypermedia

To make this discussion concrete, we define a PH application as a hypermedia application (i.e. the access to information objects is done by navigation) in which all or some of the objects of interest are real-world objects which are visited by the user "physically". The most usual scenario for these applications involves a mobile user and some location sensing mechanism and underlying software that can determine, for example, when the user is within interaction range of one of these objects. For the sake of conciseness, we also assume that digital information (data about physical objects and links) is obtained from a Web server and navigated using a browser. A simplify schema showing these ideas is presented in Figure 1.

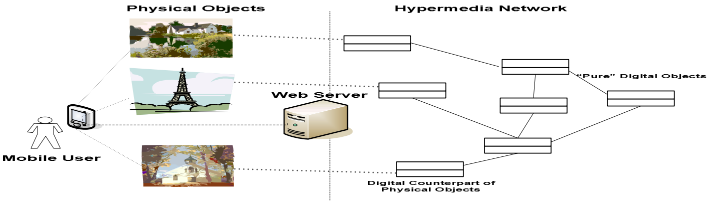


Fig. 1. Physical Hypermedia and the Web

We chose to extend the Object-Oriented Hypermedia Design Method (OOHDM) [14] by incorporating the concept of physical objects and "walking" navigation [9]. In a PH application, we aim at expressing, in an implementation-independent way, which are the objects of interest and their properties (including their location), how they are linked, which links should be implemented as conventional and which should be "walked" by the user. Following our approach, a PH application is developed in a four stages process: application modeling, navigation design, user interface design and implementation.

During application modeling we produce a two-layered model; the first layer contains the application objects, their properties, relationships with other objects and behaviors (described in UML [17]); in the second one, we describe the physical (e.g. location) counterparts of application classes. Physical Objects are described as roles, in fact decorations [15,4] of digital objects and contain the object's location, geographical relationships, and the behaviors needed to manipulate positions, such as determining if the user is in front of the object or calculating how to reach a physical object. In Figure 2, we show these two layers.

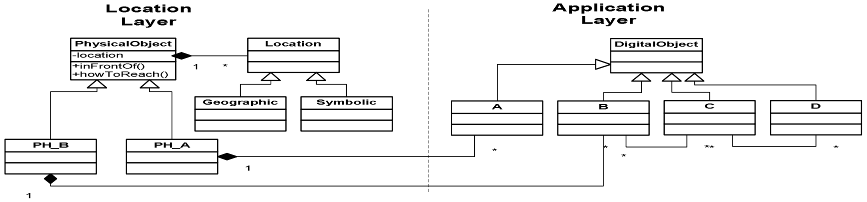


Fig. 2. The products of Application Modeling

The two physical classes (*PH_A*, and *PH_B*) wrap application classes A and B, thus obliviously adding them their physical properties. The navigation model specifies which nodes the user will explore and the links connecting these nodes. Nodes are defined as views on application objects and contain the information to be displayed. Links can be digital or physical. A digital link allows "conventional" navigation (i.e. as in the Web), while physical or "walking" links express a relationship in which the target object is physical, and exploring the object implies that the user must change his current position. Notice that physical links might be derived from both conceptual and geographical relationships. More details on the design approach can be read in [5,6].

2.2 The MVC Metaphor for Web Applications Development

The Model-View-Controller [11] is perhaps the most established paradigm for developing interactive applications. Originally developed for desktop software in the context of the Smalltalk environment, it has evolved and it is widely used in Web applications development. It proposes to partition the concerns of an interactive application in three components.

- the Model, which contains the basic application's data and behaviors.
- the View(s) which comprises the user interface objects.
- the Controller (s) which is in charge of managing user interaction, and coordinating the View and the Model.

The MVC has been implemented in different platforms and there are dozens of tools supporting software development with the MVC, for example [10]. In our case, we decide by the popular Jakarta Struts.

We chose to extend the MVC model for several reasons: first, MVC provides a reasonable model for separation of concerns in (mobile) Web applications; besides, we have used MVC-based architectures to support the implementation stage for OOHDM models [2], and finally it is a well-known metaphor, used in every modern middleware platform for Web application development. In the following sections we describe our approach, concentrating in the server-side. Details on client side adaptations such as providing communication mechanisms between sensing hardware and software and Web browsers, though important in our research, are outside the scope of this paper.

3 A MVC Framework for Physical Hypermedia

From a thorough analysis of each one of the MVC's components, we decided to extend the Controller component to support location-aware requests. We decided to do this because we found that both the View and the Model components can support Physical Hypermedia functionality, without modifying their essence.

As explained in Section 2 the Controller acts as a coordinator among the Model and the View. In our extension, the controller will need to identify if a request implies managing a location, and it will be in charge to process those requests that do involve location information.

3.1 A Conceptual View of the Location-Aware MVC

The two main components of the controller are, according to [10]: the *InputController* and the *ApplicationController*. Our extension involves the *InputController* since it deals with resolving a parameter of the request, in particular the recognition of the physical object in the user's vicinity. For the sake of modularity and compatibility we avoided changing this component but we introduced a new one, the *LocationController*. In this way we didn't clutter the standard controller with new functionality and, besides, both of them can evolve separately.

The *LocationController* will deal with those implicit or explicit parameters which correspond to requests that involve location information. Considering that the kind of pre-processing needed by a broader range of applications might involve other issues, we devised a *DispatcherController* as a Façade [4] to determine which specific controller receives control; i.e. depending on the nature of the application the *DispatcherController* establishes which Controller will be the actual *InputController*'s collaborator as shown in Figure 3. In the case of requests which do not need any pre processing, the *DispatcherController* delegate control directly to the *ApplicationController.*

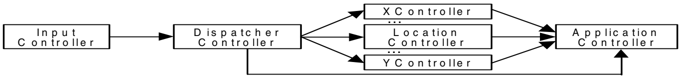


Fig. 3. Adding a new Controller

The *DispatcherController* analyzes the request. If it is a pure digital request it delegates control to the *ApplicationController.* If the request involves location information it delegates to the *LocationController*, which will analyze the location issues. A complete diagram of the extended MVC architecture is shown in Figure 4. To make the discussion more concrete we next detail our Struts implementation.

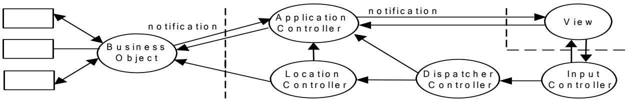


Fig. 4. Extended MVC for Physical Hypermedia

3.2 Adding Location-Awareness to Struts

By distinguishing the format of a URL contained in a request, Struts allows to define more than one control Servlet. In our implementation, a URL with the traditional format (*.do), will be dealt by the *ActionServlet.* Meanwhile, a URL with a location-compliant form (in our case /location/*), will be analyzed by the *LocationActionServlet.* This is an easy and straightforward way to implement the task of the *DispatcherController*. Both, the configuration of the new Servlet, and the format of the location-compliant URL are configured in the Struts file web.xml. We next examine how to carry out the *LocationController*'s task.

Physical Hypermedia applications may use different location models (e.g. symbolic, geometric, etc); this means that the location contained in the request has to be interpreted in the corresponding location system to obtain a correct result. To achieve this goal we decouple the corresponding functionality and create a hierarchy of *LocationFinder* classes.

LocationFinder is an abstract class which describes the common functionality of all location finders. Concrete subclasses (e.g. *SymbolicLocationFinder*) allow the developer to specialize this functionality. Sub-classes must implement at least two methods: one to identify the physical object which the user is facing (*inFrontOf*), and the other that returns the path between two physical objects (*howToReachFrom*). Each concrete sub-class implements this functionality using the concrete location model and interacting with physical (application) objects defined in the Model component.

As the objects returned by these methods must be used both by the Actions and by the JSPs (the View) we make them persistent by storing them in the Struts's session, under the name specified by the developer in the configuration file.

Following the standard way to extend Struts with specific business logics, we decided to create a specialized *RequestProcessor*, the *LocationRequestProcessor*, which is configured in the file location-struts-config.xml. The *LocationRequestProcessor* collaborates (with the mediation of the *LocationActionServlet*) with the concrete *LocationFinder* to implement the pre-processing of the request.

To complete the specification we also defined: *LocationEvent* (an obligatory property in the configuration file location-struts-config.xml, which allows to specify which of the *LocationFinder´*s method must invoke the *LocationRequestProcessor)*, *LocationActionMapping* (allows the retrieval of the new LocationEvent property), *Location-struts-config_1_1.dtd* (allows considering the incorporation of the new property)*, LocationConfigRuleSet* (incorporate the new property to the structure of the file location-struts-config.xml).

In summary the relation between the MVC elements and those that arise from the extension of Struts can be represented as shown in Figure 5:

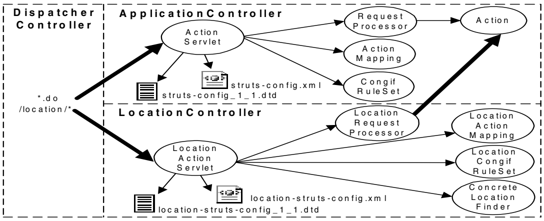


Fig. 5. The complete Struts extension for handling location data

4 Using the Framework

As a proof of concept we have instantiated the framework in a Natural Sciences Museum. Though our prototype uses a particular sensing mechanism (infrared sensors) and location model (symbolic), most design decisions can be easily understood while analyzing the example. We first produced a conceptual model, including the location enrichment. In Figure 6 we show a simplified diagram including some attributes and relationships for animals and the period in which they lived. The location attribute has been simply defined as an identifier, because we used a simple location model. Physical Animal also includes some attributes corresponding to the physical object. Objects in the conceptual model have been instantiated and mapped into a Java implementation; the specification of nodes in the navigational model were used to produce a set of JSP specifications (some of them are shown in Figure 8.a and 8.b).

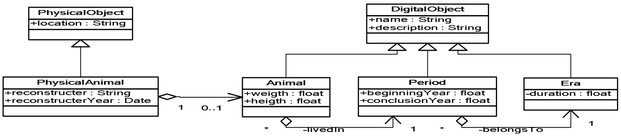


Fig. 6. Application and Physical Models for the Museum

Suppose that the user is in front of a Herrerasaurus (whose corresponding sensor emits the identifier "1"). The sequence diagram in Figure 7 shows how the process of generating the corresponding page proceeds.

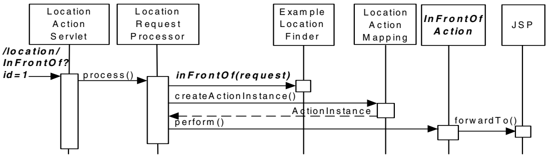
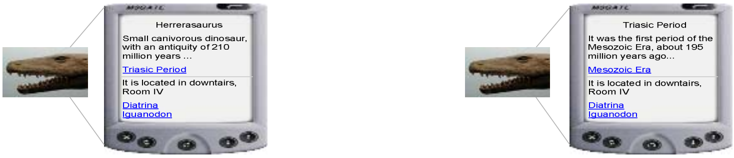


Fig. 7. Viewing information on a physical object

The *ExampleLocationFinder*, has the responsibility of finding the application objects which corresponds to the physical object with id “1”. As a response the user receives the page shown in Figure 8.a, in which digital links are in the top pane and physical information and links in the bottom pane.

We used the Builder design pattern [4] to separate the construction of the two elements of the JSP (digital and physical information); this allows us to provide digital navigation (e.g. the user clicks on “Triasic Period”) without changing the physical links exposed to the user. This means that while the user stands in front of the Herrerasaurus, the bottom pane does not change, as shown in Figure 8.b.

Physical links pose another implementation challenge. For example when the user selects “Diatrina” (another physical object in the museum), he should be instructed on the best way to reach the object, e.g. showing him a map as shown in Figure 9.



Herrerasaurus

Small canivorous dinosaur, with an antiquity of 210 million years ...

Triasic Period

It is located in downtairs, Room IV

Diatrina
Iguanodon

Triasic Period

It was the first period of the Mesozoic Era, about 195 million years ago...

Mesozoic Era

It is located in downtairs, Room IV

Diatrina
Iguanodon

Fig. 8. a: Exploring a physical object, b: Pure Digital navigation

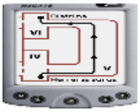


Fig. 9. The physical path to reach an object

In our extended MVC framework, the execution sequence is similar to the one in Figure 7. The only changes are the invocation of the request, the method that is executed in the *ExampleLocationFinder*, and the Action that performs the needed behavior.

We were able to develop the whole application by thinking in terms of a typical Web application, modeling it using the light extension of OOHDM and using predefined classes of our Struts extension.

5 Related Work

In [8] a comprehensive framework (HyCon) for deploying applications in which the hypermedia paradigm is extended to the physical world is presented. In [13] meanwhile, an object-oriented framework called HyperReal, based on the Dexter hypertext reference model is presented. We have followed a different strategy; instead of building a full-fledged, proprietary framework, we decided to extend a popular and widely used framework like MVC and its Struts implementation. In our first release, we decided to sacrifice facilities to keep our extension small and easy to use. The other important difference with respect to existing approaches is that our development framework is accompanied by a modeling and design approach. While the literature has focused mainly on implementation and usability issues (See [8]), we think that modeling and design aspects are critical to assure quality and quality of use.

6 Concluding Remarks and Further Work

In this paper we have presented a design and implementation framework for developing physical hypermedia applications, i.e. those applications in which physical and digital objects are related using the hypermedia paradigm.

We have shown how to slightly extend the MVC metaphor to support location-aware controllers; we have then presented a Jakarta Struts implementation of our ideas, together with a simple proof of concept for a physical hypermedia in a Natural Sciences Museum. In this implementation, we have shown how to keep physical links available while navigating digitally.

We are currently working on several research directions. One of them relates with providing better modeling and design tools to express navigational structures. We are also experiencing further navigation issues. Physical navigation introduces new possibilities such as deviating from the suggested path (e.g. as shown in Figure 9) to explore other objects. We are researching on which software support must be provided by an application framework to support the developer job in keeping track of the user trajectory, suggest possible stops, etc. We are still compromised to closely follow the MVC metaphor, even in this kind of extensions. Many of the issues discussed here have been previously explored, for example in the hypertext community [1], though standard developing tools do not exist yet. We are finally porting our implementation to the .Net platform and studying usability issues.

References

1. Adaptive Hypermedia Home Page: http://wwwis.win.tue.nl/ah/
2. M. Douglas, D. Schwabe, G. Rossi: A software arquitecture for structuring complex Web Applications. Journal of Web Engineering 1 (1): 37-60 (2002)
3. F. Espinoza, P. Persson, A. Sandin, H. Nystrom, E. Cacciatore, M. Bylund: GeoNotes: Social and Navigational Aspects of Location-Based Information Systems. Proceedings of Third International Conference on Ubiquitous Computing (Ubicomp 2001), Springer Verlag, 2-17
4. E. Gamma, R. Helm, R. Johnson, J. Vlissides: Design Patterns. Elements of reusable object-oriented software, Addison Wesley 1995
5. S. Gordillo, G. Rossi, D. Schwabe: Separation of Structural Concerns in Physical Hypermedia Models. CAiSE 2005: 446-459
6. S. Gordillo, G. Rossi, F. Lyardet: Modeling Physical Hypermedia Applications. SAINT Workshops 2005: 410-413
7. K. Gronbaek, J. Kristensen, M. Eriksen: Physical Hypermedia: Organizing Collections of Mixed Physical and Digital Material. Proceedings of the 14th. ACM International Conference of Hypertext and Hypermedia (Hypertext 2003), ACM Press, 10-19
8. F. Hansen, N. Bouvin, B. Christensen, K. Gronbaek, T. Pedersen, J. Gagach: Integrating the Web and the World: Contextual Trails on the Move. Proceedings of the 15th. ACM International Conference of Hypertext and Hypermedia (Hypertext 2004), ACM Press. 2004

9. S. Harper, C. Goble, S. Pettitt: proximity: Walking the Link. In Journal of Digital Information, Volume 5, Issue 1, Article No 236, 2004-04-07. Available at: http//jodi.ecs.soton.ac.uk/Articles/v05/i01/Harper/
10. A. Knight, N. Dai: Objects and the Web**.** IEEE Software, January/February 2002, 51-59,
11. 11.G. Krasner, S. Pope: A Cookbook for Using Model-View-Controller User Interface Paradigm in Smalltalk-80. Journal of Object Oriented Programming, August/September, 1988, 26-49.
12. OMG Model-Driven-Architecture. In http://www.omg.org/mda/
13. 13.L. Romero, N. Correia: HyperReal: A Hypermedia model for Mixed Reality. Proceedings of the 14th ACM International Conference of Hypertext and Hypermedia (Hypertext 2003), ACM Press, 2-9
14. D. Schwabe, G. Rossi: An object-oriented approach to web-based application design. Theory and Practice of Object Systems (TAPOS), Special Issue on the Internet, v. 4#4, October, 1998, 207-225.
15. F. Steimann: On the Representation of Roles in Object-Oriented and Conceptual modeling. Data and Knowledge Engineering 35 (2000) 83-106
16. The Struts Home Page: http://struts.apache.org/
17. The UML Home Page: www.omg.org/**uml/**

Replicated Ubiquitous Nets*

Fernando Rosa-Velardo, David de Frutos-Escrig, and Olga Marroquín-Alonso

Dpto. de Sistemas Informáticos y Programación,
Universidad Complutense de Madrid
{fernandorosa, defrutos, alonso}@sip.ucm.es

Abstract. In this paper we extend our basic model of Ubiquitous Nets, by adding a replication operator that creates new copies of the net firing the replicating transition. We prove that the location attribute and thus the mobility feature are not essential characteristics of the obtained model, since it is equivalent to the particular case of *Centralized Systems*, where all components are stationary and co-located in a single location. This allows us to restrict ourselves to Centralized Systems when studying the decidability of reachability and coverability properties. In this way, we prove that both reachability and coverability remain decidable. Finally, we introduce an alternative version that includes a garbage collection mechanism that allows us to remove empty nets from the state of the system. We show that in this case coverability remains decidable.

1 Introduction

In [5] we presented a formalism based on Petri Nets [4] for the study of concurrent and distributed systems in general, and of ubiquitous systems in particular [13, 8]. A system is formalized by a set of Petri nets that can perform autonomous actions, as well as movements and synchronizations with other nets. The formalism turned out to be equivalent to P/T nets [10], which gives us many decidability results, even when dealing with infinite state systems.

Intuitively, each net can be understood as a component of the system, that may act in an autonomous way. For instance, some of them can be interpreted as (possibly mobile) agents, both with reactive and proactive behaviour. However, the widely used mechanisms of cloning of agents or spawning of new agents was not yet supported. To fill this hole, in this paper we add a new primitive to our formalism, much in the flavour of replication in the π-calculus [7] or the ambient calculus [1], that has the effect of creating a new net component, with the same structure of the net invoking that primitive, initially marked in some fixed way. This is, therefore, a cloning or replication primitive, though with the help of synchronization it can be also used to implement a spawning primitive .

With the replication primitive, we have introduced in our model an extra source of infinity. Now not only may we have an unbounded number of tokens at any place, but we may also have an unbounded number of nets. However,

* Work partially supported by the Spanish projects MIDAS TIC 2003-01000 and MASTER TIC 2003-07848-C02-01.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 158–168, 2006.

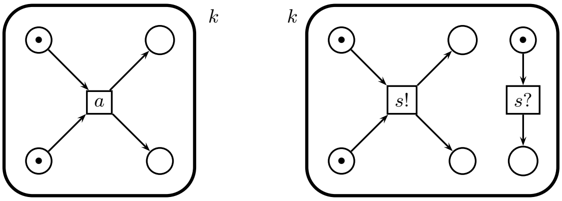


Fig. 1. Autonomous (left) and synchronizing (right) transitions

their static structure is taken from a finite set, which can be used to prove that reachability and coverability are still decidable in the extended model. Then we introduce a garbage collection mechanism, that removes from the states of our systems those nets with no tokens on them. We show that coverability is also decidable for the model with garbage collection. In the proof of both results we use *centralized systems*, which are ubiquitous systems in which all components are stationary and located within a single location. We prove that centralized systems can simulate arbitrary ubiquitous systems, so that we can restrict ourselves to them to simplify our proofs.

The rest of the paper is structured as follows. In Sect. 2 we give a brief overview of Ubiquitous Nets. In Sect. 3 we present *Replicated Ubiquitous Nets*, while an example is shown in Sect. 4. In Sect. 5 we introduce the gargabe collection mechanism. Section 6 gives the technical result of *Centralized Systems*. In Sect. 7 we show the decidability results and, finally, in Sect. 8 we present our conclusions and directions for further work.

2 Ubiquitous Nets in a Nutshell

We model ubiquitous system a collections $\mathcal{N} = \{N_1, \ldots, N_n\}$ of labelled Petri nets $N_i = (P_i, T_i, F_i, \lambda_i)$, which are located at some locality. Every net can have two types of transitions: *autonomous transitions*, and *synchronizing transitions.* Autonomous transitions are those labelled with labels in a set $\mathcal{A}$, and are as ordinary transitions in P/T nets (see Fig. 1 left). Some of the autonomous transitions are *movement transitions* labelled by *go k*, with $k \in \mathcal{L}$, the set of locality names. The firing of transitions of this kind causes the transportation of the corresponding net, whose new locality becomes k (see Fig. 2).

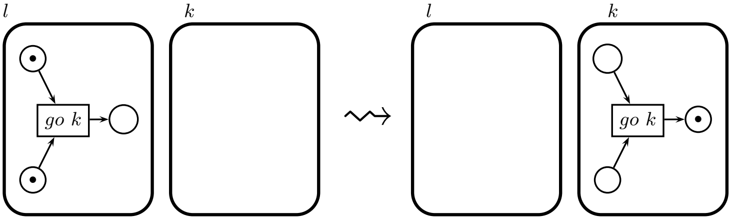


Fig. 2. Movement transitions

On the other hand, some transitions need a companion transition in the same location in order to both fire together. To be more precise, these pairs of transitions will be labelled with conjugate labels, $s?$ and $s!$, where s belongs to an alphabet of service names $\mathcal{S}$. When two compatible transitions are both enabled according to the ordinary rule in P/T nets and the nets containing those transitions are co-located then they can simultaneously fire, following the ordinary firing rule in P/T nets (see Fig. 1 right).

3 Replicated Ubiquitous Nets

In this section we formally and intuitively introduce *Replicated Ubiquitous Nets.*

Definition 1. *A* ***Replicated Ubiquitous Net*** *(RUN) is a labelled Petri net* $N = (P, T, F, \lambda)$ *where:*

- P *and* T *are disjoint sets of places and transitions.*
- $F \subseteq (P \times T) \cup (T \times P)$ *is the set of arcs of the net.*
- λ *is a function from* T *to the set* $\mathcal{A} \cup \mathcal{S}! \cup \mathcal{S}? \cup \mathcal{MS}(P)$*.*

The only difference with ordinary Ubiquitous nets is that now some transitions may be labelled by a multiset of places, that is, by a marking. That multiset corresponds to the initial marking of the replicated net that is created when firing such a transition. Marked nets are defined as usual.

Definition 2. *A marked RUN is a tuple* $N = (P, T, F, \lambda, M, k)$ *where* (P, T, F, λ) *is a RUN and* (M, k) *is a marking of* N*, that is,* M *is a multiset of places, and* $k \in \mathcal{L}$*. We will denote by* $Markings(N)$ *the set of markings of* N*.*

However, unlike for ordinary ubiquitous nets, a marking is not just a collection of individual markings, one for each net in the system, since we must also specify how many copies of each net have been created.

Definition 3. *A RUN system is a set* $\mathcal{N} = \{N_1, \ldots, N_n\}$ *of pairwise disjoint RUN's. A mapping* $\mathcal{M} : \mathcal{N} \rightarrow \bigcup_{i=1}^{n} \mathcal{MS}(Markings(N_i))$ *is a marking of* $\mathcal{N}$ *if for all* $i \in \{1, \ldots, n\}$*,* $\mathcal{M}(N_i) \in \mathcal{MS}(Markings(N_i))$*.*

Thus, $\mathcal{M}(N)$ represents the marking of the subsystem of $\mathcal{N}$ composed only of copies of N. The definitions of enabled transition and firing of transitions are analogous to those in Ubiquitous Systems. For instance, here are the definitions in the case of autonomous transitions.

Definition 4. *Given a marked RUN system* $(\mathcal{N}, \mathcal{M})$ *with* $\mathcal{N} = \{N_1, \ldots, N_n\}$ *and* $N_i = (P_i, T_i, F_i, \lambda_i)$*, we say that* $t \in T_i$ *with* $\lambda(t) \in \mathcal{A}$ *is* (M, k)*-enabled if* $(M, k) \in \mathcal{M}(N_i)$ *and* $M(p) > 0$ *for all* $p \in {}^{\bullet}t$*. The reached marking* $\mathcal{M}'$ *after the* (M, k)*-firing of* t *is defined by:*

- $\mathcal{M}'(N_j) = \mathcal{M}(N_j)$ *for every* j *with* $i \neq j$,
- $\mathcal{M}'(N_i) = (\mathcal{M}(N_i) \backslash \{(M,k)\}) \cup \{(M',\ell)\}$, *where:*
  - $M'(p) = (M(p) \backslash F(p,t)) \cup F(t,p)$ *for all* $p \in P_i$,
  - $\ell = k$ *or* $\lambda(t) = go\ \ell$.

In the previous definition, the marking of the component firing the transition is replaced in $\mathcal{N}(N)$ by the resulting marking of that firing, where N is the type of that component. Notice that, according to the last item, if the transition was labelled with *go* ℓ the location of the component is changed accordingly. Otherwise, it must be the case that $k = \ell$, that is, that the net has not moved.

The definitions of compatible synchronizing transitions and firing of a pair of compatible transitions are analogous to those in [5]. Therefore, next we just define the firing of replication transitions.

Definition 5. *Given a marked RUN system* $(\mathcal{N}, \mathcal{M})$ *with* $\mathcal{N} = \{N_1, \ldots, N_n\}$ *and* $N_i = (P_i, T_i, F_i, \lambda_i)$, *we say that* $t \in T_i$ *with* $\lambda(t) = M' \in \mathcal{MS}(P_i)$ *is* (M,k)*-enabled if* $(M,k) \in \mathcal{M}(N_i)$ *and* $M(p) > 0$ *for all* $p \in {}^\bullet t$. *The reached marking* $\mathcal{M}'$ *after the* (M,k)*-firing of* t *is defined by:*

- $\mathcal{M}'(N_j) = \mathcal{M}(N_j)$ *for every* j *with* $i \neq j$.
- $\mathcal{M}'(N_i) = (\mathcal{M}(N_i) \backslash \{(M,k)\}) \cup \{(M',k),(M'',k)\}$, *where for every* $p \in P_i$, $M''(p) = (M(p) \backslash F(p,t)) \cup F(t,p)$

Therefore, the net firing the replication transition is changed as if it were an ordinary autonomous transition. However, a new net of the same type is created, initially marked by $\lambda(t)$, and initially located wherever the net firing it is located.

In any case, we will write $\mathcal{N}(\mathcal{M})[u(M,k)\rangle\mathcal{N}(\mathcal{M}')$ if $\mathcal{M}'$ is reached from $\mathcal{M}$ after the (M,k)-firing of u, or simply $\mathcal{M}[u(M,k)\rangle\mathcal{M}'$ if there is no confusion.

4 Example: A Simple Application Scenario

This section presents a variant of the example shown in [5], that takes advantage of the new replication primitive. We model a system composed of four components, NS, TH, C and A. C and A are initially located in the same location l, while NS and TH are in different locations, k_1 and k_2 respectively (Fig. 3). Processor NS can be seen as an electronic notes system [2] that requires authentication to view its contents (action identified as service `serv2`), and TH can be seen as an electronic thermometer [12] in which the action of consulting the temperature is denoted by `serv3`. Both can also give the local time, which is denoted as service `serv1`. C is the client and A (inside the dashed line) is an agent belonging to C.

A is a process composed of two unconnected components, both within the dashed line. It is initially waiting to be told by its client to clone itself. If the client synchronizes with it in $sp1$ then it creates a new copy of itself with a token in `p`, and if it synchronizes in $sp2$ then that token is in `q` instead. In any case, the new agent can move to the designated locality and try to obtain services `serv2` or `serv3` non deterministically, trying to view the contents of the notes system

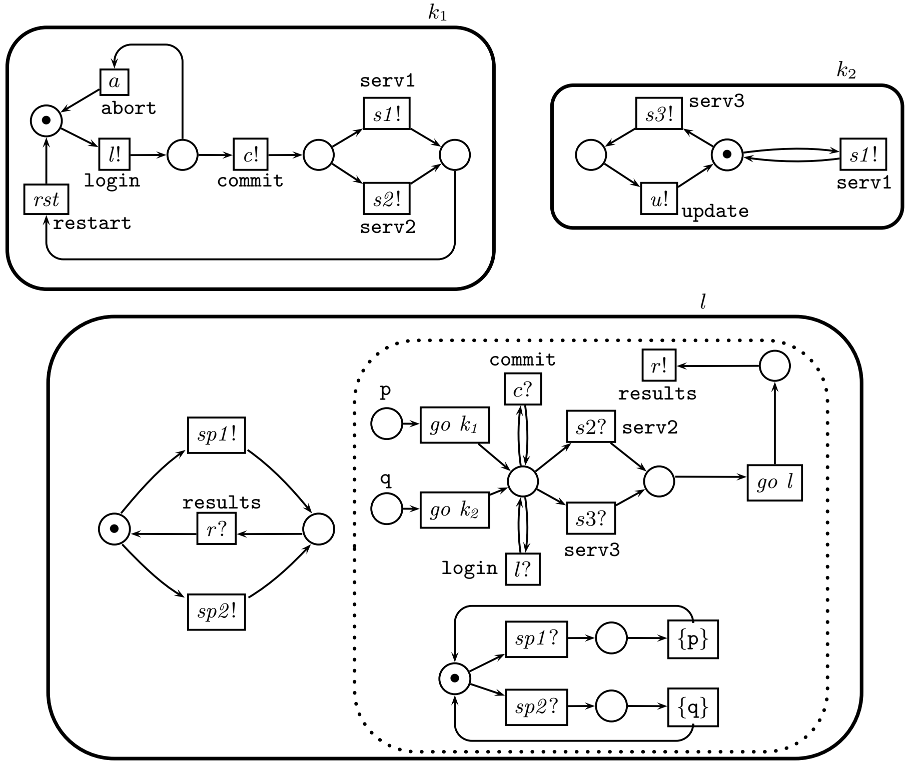


Fig. 3. An ubiquitous system modeled by Petri nets

or asking for the temperature, respectively. In the first case the note system will oblige the agent to follow an authentication protocol before being able to obtain the service (otherwise, it may abort). Note that the agent tries to obtain it even before it follows the protocol, which may be interpreted as an attempt to force the authentication. After that, the agent is willing to move back to l and offer the desired results to C.

5 Garbage Collection in RUN Systems

In the previous sections we have introduced a special type of transitions in our systems that create a component with the same structure as that where the transition is fired, but marked with a different marking, that is indicated in the (static) definition of the net. Therefore, every time one of these transitions is fired the system gains an extra component, so that systems can arbitrarily grow, as long as those transitions may be fired any number of times.

On the contrary, note that under the current definition, it is not possible for our systems to be reduced. In other words, once a net component is created it can never be removed, even if it is deadlocked. To partially avoid this problem, in this section we propose a rather conservative approach: We will consider a component net to be garbage in some marking whenever all of its places are empty. We are assuming that we have no transition without preconditions, since they are always enabled, so that any net containing them should be not considered as garbage.

Then, we can define an equivalence on markings that disregards the empty net markings (representing dead components) found on system markings.

Definition 6. *Given two markings $\mathcal{M}$ and $\mathcal{M}'$ of a RUN system $\mathcal{N}$ we will write $\mathcal{M} \equiv \mathcal{M}'$ if for all $N \in \mathcal{N}$, $\mathcal{M}(N) \equiv_N \mathcal{M}'(N)$, where $\equiv_N$ is the least equivalence relation on multisets of markings of N such that $A \equiv_N A \cup \{(\emptyset, k)\}$ for all $k \in \mathcal{L}$.*

Let us consider again the example in Sect. 4. Every time the client C synchronizes with A a new agent is spawned. Notice that after this new agent gives the results back to C it holds no tokens, so that according to the previous definition it can be considered as garbage and can be disposed, that is, removed from the marking.

6 Centralized Ubiquitous Systems

In this section we present a simulation result that we will need in the next section. First, we define a special case of RUN systems, in which every component is stationary (do not have any movement transition) and they are all at the same location. Then we prove that these assumptions do not in fact impose any real restriction, in the sense that using these particular systems we can capture the behaviour of any RUN system.

Definition 7. *A centralized Net is a Replicated Ubiquitous Net $N = (P, T, F, \lambda)$ such that $\lambda(t) \neq go\ k$ for every $k \in \mathcal{L}$ and $t \in T$.*

Definition 8. *A centralized RUN system $\mathcal{N}$ is a RUN system in which every component net is a centralized net. A mapping $\mathcal{M} : \mathcal{N} \rightarrow \bigcup_{N \in \mathcal{N}} \mathcal{MS}(Markings(N))$ is a marking (as centralized system) of $\mathcal{N}$ if there is $k \in \mathcal{L}$ such that for every $N \in \mathcal{N}, \mathcal{M}(N) \in \mathcal{MS}(Markings(N))$ and for every $(M, \ell) \in \mathcal{M}(N)$ it holds $\ell = k$.*

Notice that by definition every net in a centralized system is stationary, that is, it has no movement transitions. Moreover, when dealing with centralized systems we assume that every net is located at the same location, so that we can omit the location function from the marking of a system.

Next we describe the simulation procedure, by means of which we can encode any RUN system into a centralized one. The set L of localities appearing in the initial marking is finite. Therefore, we can add to each net N the set of places $\{N@\ell \mid \ell \in L\}$, so that a token in $N@\ell$ means that the net N is located at ℓ. In order to simulate movements we replace every transition labelled by $go\ k$ by a set of transitions labelled by $go(l, k)$, for every $l \in L$, and for each of them we add arcs arcs that remove a token from $N@l$ and place it in $N@k$. Finally, in order to have synchronizations only between co-located nets, we replace every synchronizing transition t by a set of synchronizing transitions t_l, for each $l \in L$, and for each t_l we add an arc from $N@l$ to t_l.

Theorem 1. *Every RUN system can be (strongly) simulated by a Centralized Run System.*

Therefore, centralized systems can stepwise simulate any RUN system, so that in the following we can use centralized systems instead of RUN systems when needed.

7 Verification of RUN Systems

We have defined an extension of Ubiquitous Nets, with a replication operator. This operator leads us to systems with an extra infinite dimension, since the number of nets that may compose a system is unbounded. In [10] we showed that Ubiquitous Nets were in fact equivalent to P/T nets, so that in particular both reachability and coverability were decidable [9, 4]. This equivalence does no longer seem to hold, but in the rest of the paper we prove that the main properties of these systems can still be decided. We focus on reachability and coverability, in terms of which usual safety properties can be stated.

7.1 Decidability of Reachability and Coverability in RUN Systems

In this section we show decidability of reachability and coverability for RUN systems, by reducing it to reachability and coverability in MSPN systems with naturals as identifiers [10], which is a coloured version [6] of the formalism without replication, for which we know both properties to be decidable [11]. The simulation consists on using a single net skeleton for each net type appearing in the RUN system, using identifiers in order to distinguish tokens belonging to different nets of the same type. However, in principle this would pose a technical problem, since we would need a way to distinguish between different nets of the same type located at different localities. This is why we have included in the paper the previous section: Instead of working with the original RUN system, we will work with the equivalent centralized RUN system, so that we do not need to mind about locations.

For simplicity let us assume that every net has only one copy in the initial marking and that the initial marking associated to every replicating transition t is 1-safe, that is, it has at most one token per place, which means that every p appears at most once in $\lambda(t)$. This imposes no restriction, since we can introduce

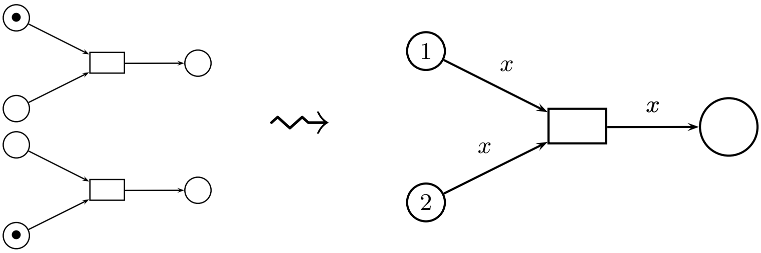


Fig. 4. Not enabled transitions

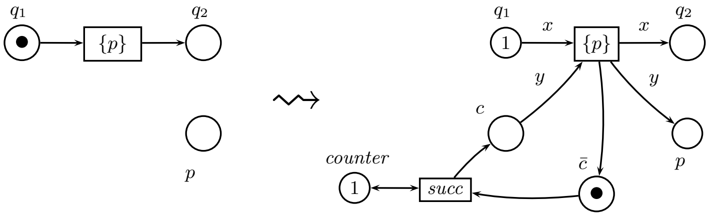


Fig. 5. Simulation of replicating transitions

in any such net a *header* by means of which any arbitrary marking can be generated starting from a 1-safe marking.

Let $\mathcal{N} = \{N_1, \ldots, N_m\}$ be a centralized RUN system. As said before, we will use each $N \in \mathcal{N}$ to simulate the behaviour of every net of type N created along the history of the system. In order to distinguish between different nets of the same type we use the tokens of the form (i, n), so that one token (i, n) represents one token in the n-th copy of N_i. In order to avoid confusion between different copies of the same net, we label all the arcs in the net with a single variable x, but using a different one for each net, so that all the tokens in precondition and postcondition places of any transitions *belong* to the same copy of the net. For instance, the marked net in Fig. 4 corresponds to a system with two copies of the same net, one of them with a token in p and the other with a token in q.

This simple construction would not work if there was any transition without preconditions, but it is not difficult to modify it in such a way that it also worked even in that case.

We simulate the creation of a new net by means of the generation of a new identifer token (i, n), which is achieved by the *succ* transitions (see Fig. 5). Any of this transitions has a distinguished precondition place called *counter* that contains a token of the form (i, m), which is replaced by another $(i, m+1)$ when the transition is fired. Moreover, the new token is copied to every identifier postcondition of the successor transition. Any replicating transition, that is just an autonomous transition in the simulating net, takes an identifier token from a precondition place c and copies it into $\lambda(t)$, thus creating the initial marking of the replicated net. Besides, it triggers the firing of the *succ* transition, so that a different identifier will be available for the next replica of the net.

In our simulation the value of each counter tells us how many copies of each net have been created. Moreover, from the marking we can also deduce the order in which they have been created, which is an information we did not have in the original markings. However, given a marking to reach (or cover) there are only a finite amount of orderings in which the nets may have replicated. Therefore, since reachability and coverability are decidable for MSPN systems we get the following

Theorem 2. *Reachability and coverability are decidable for RUN systems.*

7.2 Decidability of Coverability for RUN Systems with Garbage Collection

The construction seen in the previous section does no longer work when we consider the version of RUN systems with garbage collection. Notice that we have used natural identifiers to distinguish between the occurrences of different nets of the same type. In particular, every net has its counter place, that counts how many nets have been created (and the order in which they have been replicated). In the absence of garbage collection this was desirable, since this information is part of the markings, given that nets are created but never removed.

However, when we introduced a garbage collection mechanism, markings do not have anymore the information of how many components have been created, since any unmarked net can be removed at any time. Thus, we need a more abstract way to simulate RUN systems with garbage collection, without taking into account the number of nets that have been created.

One way of doing this is using abstract identifiers instead of natural numbers [10]. All we need to correctly simulate RUN systems is to distinguish between different copies of the same net. Thus, abstract identifiers are adequate. We proceed as in the previous section, but using abstract MSPN systems, so that, in particular, we do not need counter places anymore. The simulation works in the same way except that, when a particular identifier is removed from the net, it can be reused, thus getting the desired garbage collection mechanism.

However, reachability is not decidable for abstract MSPN systems, although coverability still is [11]. Moreover, using abstract identifiers we do not keep track of the order in which the nets have been created, so that when deciding coverability we do not need to search among the different possible orders, as we had to do in the previous section.

Theorem 3. *Coverability is decidable for RUN systems with garbage collection.*

8 Conclusions and Future Work

In this paper we have extended our model of Ubiquitous Nets with a replication operator that allow components to replicate themselves, but marked with a fixed initial way that is specified at any replicating transition. This mechanism is a simple way of formalizing the cloning of agents in a distributed environment, and can be used to implement spawning of new agents, therefore improving the usability of our formalism.

This extension adds an extra infinite dimension in the set of reachable states, since now there can be an unbounded number of net components, each with an unbounded number of tokens in their places. However, most of the interesting properties remain decidable, namely reachability and coverability of markings.

In the first model with replication presented, every time a net is created it is added to the state. In particular, in each state there are as many nets as created nets, except for those that were already in the initial marking. Therefore, states keep track of the number of times each net has been replicated. This may be an undesirable situation in certain cases, so that we have developed an alternative setting with a garbage collection mechanism. More precisely, we identify any state in which there is a net component with no tokens with the corresponding sate where this net has been removed. We have proved that coverability is still decidable in the presence of garbage collection, although we do not know yet if this is also the case for reachability.

In the proof of the previous results we made use of the analogous results in [11], but in order to obtain the simulations we had to work with an equivalent formalism to RUN systems, namely that in which components do not move. We have defined this apparent particular case of systems and prove that, in fact, it is equivalent to the whole class of RUN systems.

Due to lack of space, we have only considered the extension of the basic model, without locations or any other kind of identifiers. In the first case, the results proved here would still hold, and all the proofs are conceptually similar, although technically a bit more involved. However, the case of MSPN systems with identifiers has proved to be rather more complicated and we are currently studying it in detail.

Finally, we have developed a tool for the integrated design and verification of systems based on these nets, implemented in the reflective programming language Maude [3], based on rewriting logic. We intend to extend this tool with the replication operator presented in this paper.

References

[1] L. Cardelli and A. D. Gordon. *Mobile ambients.* In *Foundations of Software Science and Computation Structures: First International Conference, FOSSACS'98*, volume 1378 of *LNCS*, pages 140–155. Springer, 1998.

[2] K. Cheverst, A. Dix, D. Fitton and M. Rouncefield. *'Out To Lunch': Exploring the Sharing of Personal Context through Office Door Displays.* Proceedings of the Australasian Computer-Human Conference-OzCHI 2003, pp.74-83. 2003.

[3] M. Clavel, F. Durán, S. Eker, P. Lincoln, N. Martí-Oliet, J. Meseguer and C. Talcott. The Maude 2.0 System. In Proc. Rewriting Techniques and Applications, 2003. LNCS vol. 2706, pp. 76–87. Springer-Verlag, 2003.

[4] Jörg Desel and Wolfgang Reisig. *Place/transition petri nets.* Lectures on Petri Nets I: Basic Models, LNCS vol.1491, pp.122–173. Springer-Verlag, 1998.

[5] D. Frutos Escrig, O. Marroquín Alonso and F. Rosa Velardo. *Ubiquitous Systems and Petri Nets.* Ubiquitous Web Systems and Intelligence, LNCS vol.3841. Springer-Verlag, 2005.

[6] K. Jensen. *Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use..* Volume 1, Basic Concepts. Monographs in Theoretical Computer Science, Springer-Verlag, 2nd corrected printing 1997. ISBN: 3-540-60943-1.

[7] R. Milner, J. Parrow and D. Walder. *A Calculus of Mobile Processes I.* In Information and Computation, vol.100(1)1-40. Academic Press Inc., 1992.

[8] R. Milner. *Theories for the Global Ubiquitous Computer.* Foundations of Software Science and Computation Structures-FoSSaCS 2004, LNCS vol.2987, pp.5–11. Springer-Verlag, 2004.

[9] C. Reutenauer. The Mathematics of Petri Nets. Masson and Prentice Hall, 1990.

[10] F. Rosa Velardo, D. Frutos Escrig, and O. Marroquín Alonso. *Mobile Synchronizing Petri Nets: a choreographic approach for coordination in Ubiquitous Systems.* In 1st Int. Workshop on Methods and Tools for Coordinating Concurrent, Distributed and Mobile Systems, MTCoord'05. ENTCS (to appear).

[11] F. Rosa Velardo, D. Frutos Escrig, and O. Marroquín Alonso. *On the expressiveness of Mobile Synchronizing Petri Nets.* In 3rd International Workshop on Security Issues in Concurrency, SecCo'05. ENTCS (to appear).

[12] R. Want. *Enabling Ubiquitous Sensing with RFID.* Computer vol.37(4), pp.84-86. IEEE Computer Society Press, 2004.

[13] M. Weiser. *Some Computer Science Issues in Ubiquitous Computing.* Comm. of the ACM vol.36(7), pp.74-84. ACM Press, 1993.

Design of a Shared Ontology Used for Translating Negotiation Primitives

Joaquín Pérez[1], Maricela Bravo[1], Rodolfo Pazos[1], Gerardo Reyes[1], Juan Frausto[2], Víctor Sosa[1], and Máximo López[1]

[1] Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET),
Cuernavaca, 62490, México
{jperez, mari_clau, pazos, greyes, vjsosa, maximo}@cenidet.edu.mx

[2] ITESM, Campus Cuernavaca, 62589, México
juan.frausto@itesm.mx

Abstract. In this paper we present the design of a shared ontology, with the objective to translate a variety of negotiation primitives. Our approach focuses on facilitating communication among agents during negotiation process execution. Traditional negotiation systems impose several restrictions on the type and format of negotiation primitives that can be exchanged among agents. In contrast, we propose the incorporation of an ontology-based solution to overcome heterogeneity and provide communication facilities for participation in negotiations based in open environments such as Internet. To evaluate our ontology we implemented a Web service-oriented negotiation system, and incorporated a translation module that uses the ontology as a vocabulary of negotiation primitives. The experimental results show that the incorporation of the ontology improves the continuity of the execution of negotiation processes, resulting in more agreements.

1 Introduction

Negotiation has been a central topic mostly stressed in distributed artificial intelligence. But recently has gained special interest the execution of negotiation processes over Internet. Negotiation is a process in which two or more software agents interact and take decisions for mutual gain. This interaction is executed through the exchange of negotiation primitives among agents, during this interaction each agent uses its own local language for formulating messages according to their needs and plans.

Traditional negotiation systems impose several limitations on the type and format of negotiation primitives. For example, if a new agent wants to participate in a negotiation process, it has to be redesigned according to the protocol and language specifications. Instead, in this work we are presenting the design and development of an ontology solution to classify and explicitly describe negotiation primitives in a machine interpretable form, allowing agents to participate in negotiations over Internet, with no language restrictions. Additionally, the ontology represents a shared vocabulary of the negotiation primitives used by each agent participating in the negotiation process. To evaluate the ontology we have implemented a Web service-oriented negotiation

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 169 – 178, 2006.

system, into which we have incorporated a translator module which uses the ontology to solve language heterogeneity among agents during negotiation run time.

The rest of the document is organized as follows. In section 2, we present the methodology for building the ontology. In section 3, we describe the negotiation system architecture. In section 4 we show the results of experiments. Finally in section 5, we present conclusions.

2 A Methodology for Building the Ontology

There are some methodologies for constructing ontologies reported in literature. Three of the earlier methodologies presented were the Uschold and King [1], the Gruninger and Fox [2], and the Gómez-Pérez METHONTOLOGY [3]. We selected the methodology proposed by Uschold as our guide. This methodology establishes three general steps for building ontologies:

1. *Purpose of the ontology*. The objective of the negotiation ontology is to serve as an inter-lingua between agents during exchange of negotiation messages. The ontology is an important part of a larger electronic commerce project, which is integrated by software agents trading over Internet. Thus the users of the ontology are developers, managers and end user applications.
2. *Building the ontology*. The first step of building the ontology is *capture*. The ontology capture is the process of identifying and defining the key concepts and relationships in the domain of interest. We selected the main elements of negotiations presented by Jürgen Müller [4] as the key concepts. In his work of negotiation principles he states that the main elements of electronic negotiations are language, process and decision (see Fig. 1).

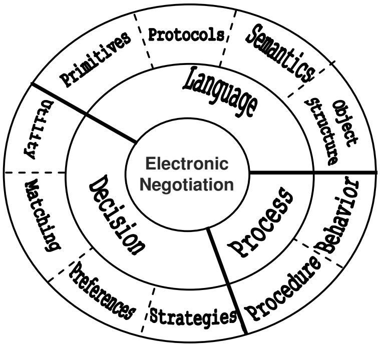


Fig. 1. Negotiation categories proposed by Jürgen Müller

We selected the language category as a main class. The language category is concerned with the communication primitives for negotiation, their semantics and their usage in terms of a negotiation protocol [4]. The classification of negotiation language primitives is divided into three groups: *initiators*, if they initiate a negotiation, *reactors*, if they react on a given statement and *completers*, whether they complete a negotiation. This is the basic structure of our ontology classification. Table 1 shows the key concepts of our ontology and a brief description of their meaning, identifying their type as classes or subclasses.

Table 1. Identification of the key concepts of the Ontology

Term	Description	Type
Language	Communication primitives for negotiation, their semantics and their usage.	Class
Decision	Algorithms to compare the negotiation topics and correlation functions.	Class
Protocols	Models of the negotiation process and the global behavior of the participants.	Class
Participants	Identifies the negotiating agents which participate in the negotiation process.	Class
Primitives	The basic form of communication between agents.	Subclass
Parameters	The context transmitted together with a negotiation primitive.	Subclass
Initiator	Negotiation primitives that initiate a negotiation.	Subclass
Reactor	Negotiation primitives that react on a given statement.	Subclass
Completer	Negotiation primitives that complete a negotiation.	Subclass
Deal	Negotiation primitives that complete a negotiation with an agreement.	Subclass
No-deal	Negotiation primitives that complete a negotiation with no deal or failure.	Subclass

We organized the concepts identified above on a hierarchical classification scheme. Fig. 2 shows the general classification of the ontology.

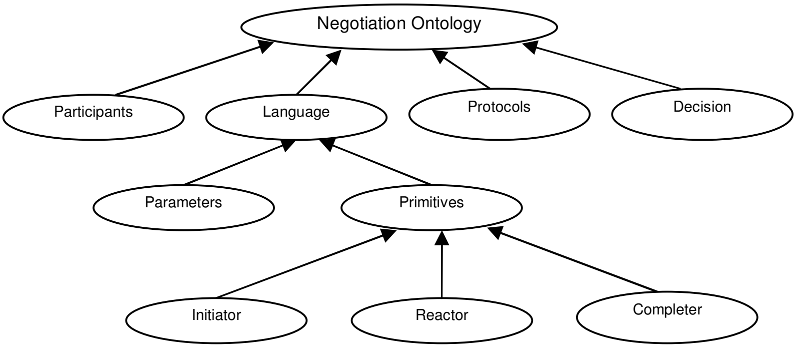


Fig. 2. General classification of the Ontology

The central topic of our ontology is the language of negotiation, for this reason we searched for the negotiation systems reported in literature, specifically we investigated the reported works about the use and definition of negotiation language. A summary of these negotiation systems with their negotiation primitives is shown in table 2.

Table 2. Negotiation primitives used in different systems

Author	Negotiation Primitives
Jin Baek Kim, Arie Segev [5]	Initial_offer RFQ Accept Reject Offer Counter-offer
Stanley Y. W. Su, Chunbo Huang, Joachim Hammer [6], Patrick C. K. Hung [7]	CFP Propose Accept Terminate Reject Acknowledge Modify Withdraw
Anthony Chavez, Pattie Maes [8]	accept-offer?(agent, from-agent, offer) what-is-price?(agent, from-agent) what-is-item?(agent, from-agent) add-sell-agent add-buy-agent add-potential-customers(sell-agent, potential-customers) add-potential-sellers(buy-agent, potential-sellers) agent-terminated(marketplace, agent) deal-made(marketplace, sell-agent, buy-agent, item, price)

Sonia V. Rueda, Alejandro J. García, Guillermo R. Simari [9]	Requests_Add(s, h, p)
	Authorize_Add(s, h, p)
	Require(s, h, p)
	Demand(s, h, p)
	Accept(s, h, p)
	Reject(s, h, p)
	Unable(s, h, p)
	Require-for(s, h, p, q)
	Insist_for(s, h, p, q)
	Demand_for(s, h, p, q)

Haifei Li, Chunbo Huang and Stanley Y.W Su [10]	Call for proposal Propose proposal Reject proposal Withdraw proposal Accept proposal Modify proposal Acknowledge message Terminate negotiation
Dignum, Jan Dietz, Egon Verharen and Hans Weigand [11]	request-quotation give-quotation order delivered paid
FIPA Communicative Acts [12]	Accept Proposal Agree Cancel Call for Proposal Confirm Disconfirm Failure Inform Inform If Inform Ref Not Understood Propagate Propose Proxy Query If Query Ref Refuse Reject Proposal Request Request When Request Whenever Subscribe

Based on the key concepts presented in table 1 and the classification of the negotiation primitives presented in Fig. 2, we built a first version of the ontology, which was populated with the negotiation primitives from table 2. To code the ontology we decided to use OWL as the ontological language, because it is the most recent development in standard ontology languages from the World Wide Web Consortium (W3C)[1]. An OWL ontology consists of classes, properties and individuals. Classes are sets that contain individuals. Properties are binary relations on individuals, they are also known as roles in description logics. Individuals represent objects or instances in the domain of interest. We developed the ontology using Protégé [13, 14], an open platform for ontology modeling and knowledge acquisition. Protégé has an OWL Plugin, which can be used to edit OWL ontologies, to access description logic reasoners, and to acquire instances of semantic markup. Fig. 3 shows part of the ontology code generated with Protégé.

```
<owl:Class rdf:ID=“Participants">
  <rdfs:subClassOf rdf:resource="#negotiation"/>
</owl:Class>
<owl:Class rdf:ID=“Language">
  <rdfs:subClassOf rdf:resource="#negotiation"/>
</owl:Class>
<owl:Class rdf:ID=“Protocol">
  <rdfs:subClassOf rdf:resource="#negotiation"/>
</owl:Class>
<owl:Class rdf:about="#MessagePrimitives">
 <rdfs:subClassOf rdf:resource="#Language"/>
</owl:Class>
<owl:Class rdf:about="#MessageContents">
 <rdfs:subClassOf rdf:resource="#Language"/>
</owl:Class>
<owl:Class rdf:ID="Initiator">
  <rdfs:subClassOf rdf:resource="#MessagePrimitives"/>
</owl:Class>
<owl:Class rdf:ID="Reactor">
  <rdfs:subClassOf rdf:resource="#primitives"/>
</owl:Class>
<owl:Class rdf:about="#Completer">
  <rdfs:subClassOf rdf:resource="#primitives"/>
</owl:Class>
<owl:ObjectProperty rdf:ID="isSuccessorOf">
  <owl:inverseOf rdf:resource=“#hasSuccesor"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#hasSuccesor">
  <owl:inverseOf rdf:resource="#isSuccessorOf"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="isSynonymOf">
  <owl:inverseOf rdf:resource=“#hasSynonym"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#hasSynonym">
  <owl:inverseOf rdf:resource="#isSynonymOf"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:ID="hasAntecessor">
  <owl:inverseOf rdf:resource=“#isAntecessorOf"/>
</owl:ObjectProperty>
<owl:ObjectProperty rdf:about="#isAntecessorOf">
  <owl:inverseOf rdf:resource="#hasAntecessor"/>
</owl:ObjectProperty>
```

Fig. 3. Part of the ontology code generated with Protégé

3. *Evaluation of the ontology*. Citing Gómez-Pérez [3], evaluation refers to the technical judgment of the ontology, their associated software environments and documentation with respect to a frame or reference. The frame or reference may be requirements, specifications, competency questions or the real world. We evaluated the ontology into the application to check if it satisfied the purpose of developing it.

[1] http://www.w3.org

In section 3 we describe the negotiation system architecture which uses the ontology as an inter-lingua to translate negotiation primitives. In section 4 we show the experimental results of the negotiation system with and without the translator module.

3 Negotiation System Architecture

The system architecture for executing negotiation processes over Internet is illustrated in Fig. 4. This architecture is integrated by a matchmaker module, a negotiation process module and a translator module.

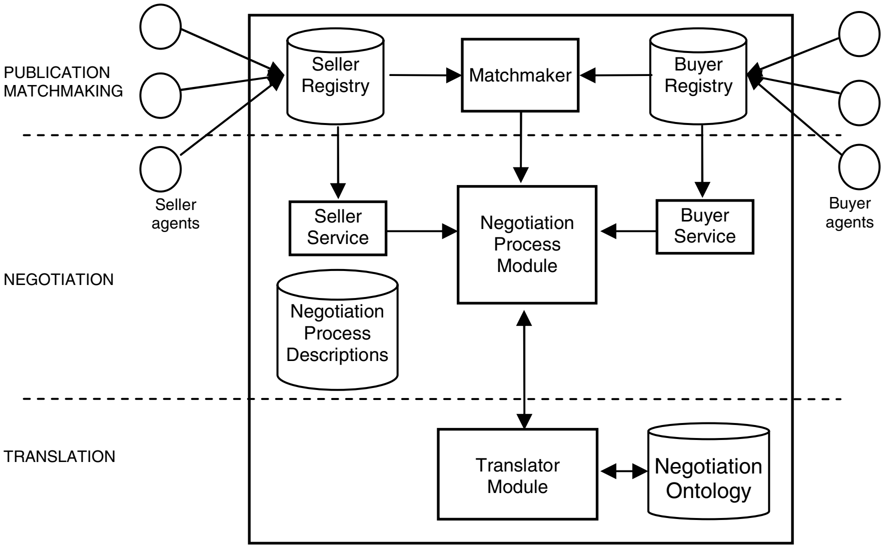


Fig. 4. General architecture for execution of negotiation processes

The matchmaker module is continuously browsing buyer registries and seller descriptions, searching for coincidences. The negotiation process module is responsible for controlling the execution of negotiation processes among agents according to the protocol. The translator module is invoked whenever the agent does not know a negotiation primitive sent by the partner.

This architecture was designed following the service oriented architecture and implemented using Web service technologies. The translator module was implemented using Jena[2], a framework for building Semantic Web applications. It provides a programmatic environment for OWL, including a rule-based inference engine.

3.1 Translator Module

The translator module acts as an interpreter of different negotiation agents. In Fig. 5, we present the architectural elements involved in translation. This module consists of

[2] http://jena.sourceforge.net

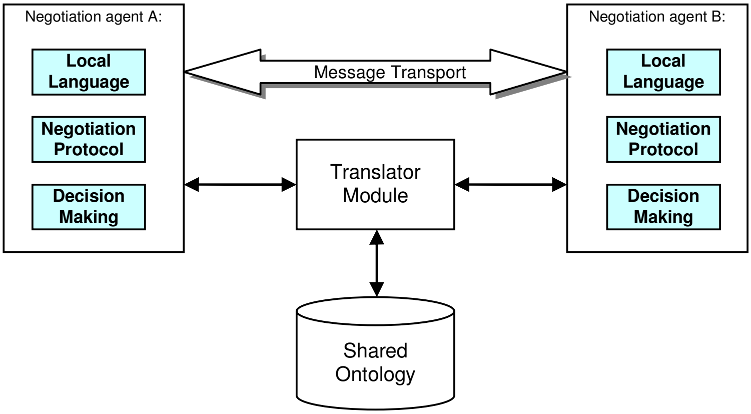


Fig. 5. Translator module architecture

multiple negotiation agents, a message transport, and the shared ontology. Each negotiation agent in turn consists of a local language, decision making strategies to determine the preferences, and the negotiation protocol.

For example, suppose that agents *A* and *B* initiate a negotiation process, using their own local language, sending messages over the message transport. If happens that agent *A* misunderstands a message from agent *B*, it invokes the translator module sending the message parameters (sender, receiver, message). The translator interprets the message based on the definitions of the sender agent and converts the message into an interlingua. Then the translator converts the interlingua representation to the target language based on the receiver agent definitions. Finally, the translator sends back the message to the invoking agent *A* and continue with execution of negotiation.

The translator module is invoked only in the occurrence of a misunderstanding, assuring interoperability at run time.

4 Experimental Results

The negotiation experiments were executed in two phases. The first execution tested the interaction among buyer and seller agents, incorporating tests with different negotiation primitives. For the second execution we used the same strategies, and input data, but incorporated the translator module. The results of these experiments were registered in a log file. Table 3 shows the results of both cases.

The first phase results showed that it is possible to end the negotiation process with no agreement. This is mainly due to the private strategies defined inside the agents, but there is another interesting result, that is, negotiation process can end without agreement due to lack of understanding of negotiation messages.

The second phase results showed a reduction in the number of negotiations finished by lack of understanding, which does not mean that the incorporation of a translator module will ensure an agreement; but at least, the negotiation process will continue executing. Fig. 6 shows a comparison for the two phases executed.

Table 3. Negotiation results

Last price	Max pay	Rounds	Qty	Final price	1st execution	2nd execution
$ 1,750.00	$ 849.00	12	847	$ -	no offer	no offer
$ 774.00	$ 1,760.00	3	887	$ 1,674.00	offer accepted	offer accepted
$ 1,788.00	$ 128.00	12	1660	$ -	no offer	no offer
$ 1,058.00	$ 110.00	12	1270	$ -	no offer	no offer
$ 694.00	$ 938.00	10	950	$ 894.00	offer accepted	offer accepted
$ 761.00	$ 77.00	12	1475	$ -	no offer	no offer
$ 1,940.00	$ 2,233.00	10	570	$ 2,140.00	offer accepted	offer accepted
$ 621.00	$ 446.00	12	56	$ -	no offer	no offer
$ 1,008.00	$ 1,235.00	10	30	$ 1,208.00	offer accepted	offer accepted
$ 114.00	$ 704.00	7	8	$ 614.00	offer accepted	offer accepted
$ 1,837.00	$ 2,199.00	9	53	$ 2,137.00	offer accepted	offer accepted
$ 1,665.00	$ 2,047.00	9	56	$ 1,965.00	offer accepted	offer accepted
$ 1,377.00	$ 1,783.00	8	31	$ 1,777.00	offer accepted	offer accepted
$ 1,920.00	$ 286.00	12	81	$ -	no offer	no offer
$ 172.00	$ 1,553.00	2	41	$ 1,172.00	offer accepted	offer accepted
$ 980.00	$ 1,541.00	2	67	$ -	**not understood**	offer accepted
$ 1,826.00	$ 2,464.00	2	99	$ -	**not understood**	offer accepted
$ 1,276.00	$ 500.00	2	43	$ -	**not understood**	no offer
$ 1,500.00	$ 1,108.00	2	110	$ -	**not understood**	no offer
$ 1,400.00	$ 1,520.00	3	4	$ -	**not understood**	offer accepted

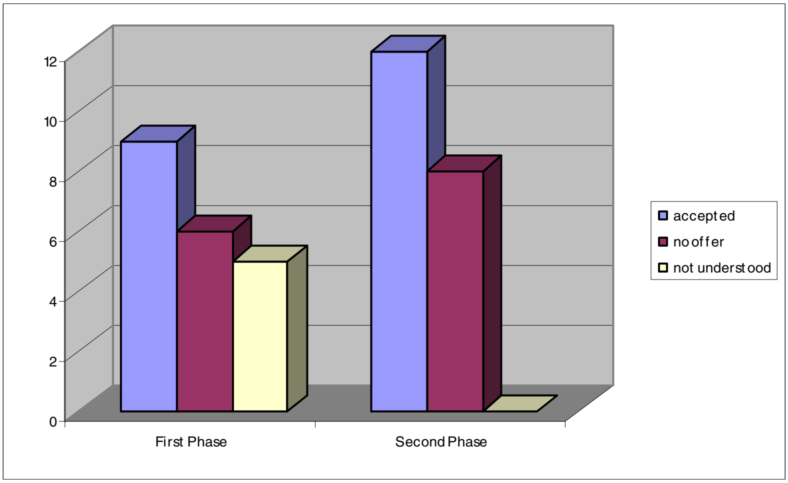


Fig. 6. Graphical comparison that shows a reduction during the second phase in the number of negotiations finished by lack of understanding

5 Conclusions

In this paper we have presented an ontology-based solution to improve the execution of negotiation processes over Internet. In particular we have incorporated a translator module for the problem of lack of understanding among seller and buyer agents during the exchange of messages in a negotiation process.

We presented the methodological steps we followed for developing the negotiation ontology. In particular, we showed the knowledge sources taken from different authors that were the basis for the identification of the key classes and individuals of the ontology. We evaluated the ontology in the target application, showing an improvement of negotiation process execution. We presented the system architecture into which we have incorporated the translator and the ontology. The experimental tests showed that the integrated architecture improves the continuity of the execution of negotiation processes, resulting in more agreements.

We believe that semantic interoperability of messages is an important issue that can be solved by incorporating ontology-based solutions.

References

1. Uschold, M. and King M., Towards a Methodology for Building Ontologies, *Workshop on Basic Ontological Issues in Knowledge Sharing*, 1995.
2. Grüninger, M. and Fox, M., The Role of Competency Questions in Enterprise Engineering, *IFIP WG 5.7 Workshop on Benchmarking*. Theory and Practice, Trondheim, Norway, 1994.
3. Fernández, M., Gómez-Pérez, A., and Juristo, N., METHONTOLOGY: From Onthological Art towards Ontological Engineering, *Proceedings of AAAI Spring Symposium Series*, AAAI Press, Menlo Park, Calif., pp. 33-40, 1997.
4. Müller, H. J., Negotiation Principles, *Foundations of Distributed Artificial Intelligence*, in G.M.P. O´Hare, and N.R. Jennings, New York: John Wiley & Sons.
5. Jin Baek Kim, Arie Segev, A Framework for Dynamic eBusiness Negotiation Processes, *Proceedings of IEEE Conference on E-Commerce*, New Port Beach, USA, 2003.
6. Stanley Y. W. Su, Chunbo Huang, Joachim Hammer, Yihua Huang, Haifei Li, Liu Wang, Youzhong Liu, Charnyote Pluempitiwiriyawej, Minsoo Lee and Herman Lam, An Internet-Based Negotiation Server For E-Commerce, *the VLDB Journal*, Vol. 10, No. 1, pp. 72-90, 2001.
7. Patrick C. K. Hung, WS-Negotiation: An Overview of Research Issues, *IEEE Thirty-Seventh Hawaii International Conference on System Sciences* (HICSS-37), Big Island, Hawaii, USA, January 5-8, 2004.
8. Anthony Chavez, Pattie Maes, Kasbah: An Agent Marketplace for Buying and Selling Goods, *Proceedings of the First International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology*, London, UK, April 1996.
9. Sonia V. Rueda, Alejandro J. García, Guillermo R. Simari, Argument-based Negotiation among BDI Agents, *Computer Science & Technology*, 2(7), 2002.
10. Haifei Li, Chunbo Huang and Stanley Y.W Su, Design and Implementation of Business Objects for Automated Business Negotiations, *Group Decision and Negotiation*, Vol. 11; Part 1, pp. 23-44, 2002.

11. Dignum, Jan Dietz, Communication Modeling – The language/Action Perspective, *Proceedings of the Second International Workshop on Communication Modeling*, Computer Science Reports, Eindhoven University of Technology, 1997.
12. FIPA Communicative Acts, http://www.fipa.org/specs/fipa00037/XC00037G.html.
13. J. Gennari, M. Musen, R. Fergerson, W. Grosso, M. Crubézy, H. Eriksson, N. Noy, and S. Tu: The evolution of Protégé-2000: An environment for knowledge-based systems development, *International Journal of Human-Computer Studies*, 58(1): 89-123, 2003.
14. H. Knublauch: An AI tool for the real world: Knowledge modeling with Protégé, *JavaWorld*, June 20, 2003.

A Web Page Ranking Method by Analyzing Hyperlink Structure and K-Elements

Jun Lai[1], Ben Soh[2] and Chai Fei[3]

[1] Department of Computer Science and Computer Engineering,
LaTrobe University
Bundoora, Melbourne, VIC 3086 Australia
jun@cs.latrobe.edu.au
[2] Department of Computer Science and Computer Engineering,
LaTrobe University
Bundoora, Melbourne, VIC 3086 Australia
ben@cs.latrobe.edu.au
[3] Beijing Army General Hospital,
Beijing, China
chaifei@sina.com

Abstract. The tremendous growth of the web has created challenges for the search engine technology. In this paper we propose a method for information retrieval and web page ranking by analyzing hyperlink structure on the web graph and the weight of keywords. Hyperlink structure analysis measures page importance by calculating the page weight based on links. This method is not counting links from all pages equally, but by normalizing the number of links on a page. The weight of keywords is computed from the elements, keywords and anchors, which we call K-elements. A linear combination of the hyperlink structure and the weight of keywords is proposed and evaluated to rank web pages. In the evaluation, we take into consideration both the importance and relevance of a page.

Keywords: Information retrieval, search engine, web crawlers, hyperlinks, elements, keywords, World Wide Web.

1 Introduction

The rapid growth of online information creates challenges for information retrieval. People are usually surfing the web using its link graph, starting with high quality human maintained indices or with search engines, which mostly rank pages by analyzing the hyperlink structure. In recent years, several information retrieval methods using the information about the link structure have been developed and proved to provide significant enhancement to the performance of Web search. The PageRank [1] and HITS [2] are two pioneers among them. Lawrence Page and Sergey Brin proposed PageRank [1], which ranks web pages based on the link structure of the web. PageRank uses simple citation theory, in which highly linked pages are more important

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 179 – 186, 2006.

than pages with few links. Pages with a link from the important page contain valuable information. Hyperlink Induced Topic Search (HITS) is introduced by Jon Kleinberg [2]. HITS classifies pages into two types, “hub pages” and “authority pages”, which mutually reinforce each other. Since then, the hyperlink analysis has been well adopted in the web community [3][4][5][6].

However, the relevance between a page and query is also very important. Anchor text is discussed in [7], where the text of a link is associated with the page the link points to. This is based on the fact that anchors often provide more accurate descriptions of web pages than the pages themselves. This idea of anchor-text first emerged and was implemented in the World Wide Web Worm [8]. Automatic Resource Compilation (ACR) [5] extends anchor text by using the text surrounding links in addition to the link text itself.

All these anchor text related researchers are trying to obtain the description of a page from a different angle of the page content. From this point of view, we propose an approach to describe a page with association of keywords and the weight of keywords based on anchor text and its content, where the weight of keywords is computed by our proposed algorithm based on the frequency of keywords, the weight of elements and anchor text. This method is called K-elements. The value of page weight measures page importance by analyzing hyperlinks structure, where the page weight is calculated iteratively based on in-coming and out-going links. Then pages are re-ranked with the linear combination of page weight and k-elements.

This paper is organized as follows: in section 2, we give the hyperlink structure based page weight measurement. Then K-elements method is discussed in section 3, followed by section 4 is the re-ranking method and evaluation. Finally, the conclusions are drawn and future work is discussed in section 5.

2 Hyperlink Structure Based Page Rank

Hyperlinks contain valuable information, which is the main difference between text documents and web pages. In addition, hyperlinks are significantly different from academic citation. Furthermore, academic papers are closely reviewed and cited for the purpose of exchanging knowledge and ideas, while hyperlinks vary in quality, usage and control. Therefore, hyperlink analysis is not just simply counting the sum of the number of hyperlinks equally weighed. In this section, we first analyze the structure of web graph in section 2.1. Then page weight measurement is proposed in section 2.2.

2.1 Hyperlink Structure

We can view the web as a directed graph, which is composed of vertices and edges [1][2][3][10]. Where vertices are web pages and edges are links between web pages. Figure 1 shows a directed web graph. We define this graph as G = (V, E), where V represents vertices in the graph, a link $l(i,j) \in$ E indicates the link between page i and page j. $l_{i->j}$ is a directed link from page i to page j.

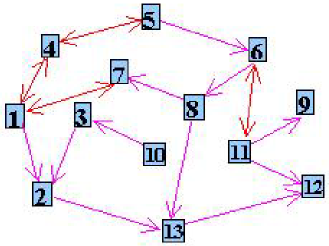


Fig. 1. Directed web graph

The web graph consists of three types of pages: hub, head and tail pages.

- Hub pages are those that have many incoming and outgoing links and playing an important multi-junction role in the traffic of web.
- Head pages only contain outgoing links, but do not have incoming links.
- Tail pages have incoming links and do not have outgoing links.

Generally speaking, highly linked pages are more important than pages with few links. A page with a link coming from a very important page should be ranked higher than those pages with links from obscure places [1]. Simple citation counting has been used to speculate on the future winners of the Nobel Prize [13].

2.2 Page Weight Measurement

In order to measure the importance of web pages, we adopt page weight measurement based on the fact addressed previously in that incoming and outgoing links define the importance of a page. By doing so, we define that P_i^{in} and P_i^{out} are weight vectors of page V_i, where out-weight P_i^{in} measures the importance of V_i in terms of incoming links and in-weight P_i^{out} measures the importance of V_i in relation to outgoing links. Let N_j^{in} be the total number of incoming links in page V_j and N_j^{out} be the total number of outgoing links in page V_j.

Each page has an initial weighting value as 1. Then P_i^{in} is the yield of P_j^{out} over the total number of outgoing links N_j^{out} in page V_j, where P_j^{out} is the out-weight vector of page V_j and we have $l_{j\to i}$. Likewise, P_i^{out} is the yield of P_j^{in} over the total number of incoming links N_j^{in} in page V_j, where we have $l_{i\to j}$. P_i^{in} and P_i^{out} are defined as follows:

$$\forall i, P_i^{in} = \sum_{l_{j\to i}} P_j^{out} \Big/ N_j^{out} \qquad (1)$$

$$\forall i, P_i^{out} = \sum_{l_{i\to j}} P_j^{in} \Big/ N_j^{in} \qquad (2)$$

The iteration continues until a fixed point reaches, where the weighting value of each page becomes stable. This iterative algorithm shows that hyperlink analysis is not simply counting the number of links in each page. In is determined by the weight of pages it connects to and the total number of links on those pages. However, for those head pages, there is only P_i^{out} and $P_i^{in} \to 0$. While for the tail pages, $P_i^{out} \to 0$. In order to measure the importance of all type of pages in the Web graph, we introduce the average in-weight $\overline{P^{in}}$ and out-weight $\overline{P^{out}}$, which are defined as follows:

$$\overline{P^{in}} = \sum_{i=1}^{n} \frac{P_i^{in}}{n} \quad , \quad \overline{P^{out}} = \sum_{i=1}^{n} \frac{P_i^{out}}{n} \tag{3}$$

where n is the total number of pages in the concerned web graph. Based on N_j^{in} and N_j^{out}, we define the weight of page P_i as follows:

$$P_i = \frac{P_i^{in} - \overline{P^{in}}}{\sigma_{in}} + \frac{P_i^{out} - \overline{P^{out}}}{\sigma_{out}} \tag{4}$$

where σ_{in} and σ_{out} are standard deviations for normalization and defined as follows:

$$\sigma_{in} = \sqrt{\sum_{i=1}^{n} (P_i^{in} - \overline{P^{in}})^2 / n} \quad , \quad \sigma_{out} = \sqrt{\sum_{i=1}^{n} (P_i^{out} - \overline{P^{out}})^2 / n}$$

To simplify the calculation of the weight of a page, we use non-linear mapping through Sigmoidal function with coefficient β, which is used in neural network to map the value of P_i in the equation (8) to the range between 0 and 1 [14].

$$\varphi_{(P_i)} = \frac{1}{1 + e^{-\beta \cdot P_i}} \tag{5}$$

3 K-Elements

By analyzing hyperlink structure, we obtain the page weight for ranking web pages. However, this page weight based on hyperlinks is an absolute value instead of relative value in terms of different keywords. To this end we introduce the second part of the proposed approach: K-elements that carry out page weights in terms of different keywords.

Currently, there are a number of metadata standards proposed for web pages. Among them are two well-publicized, solid efforts: the Dublin Core Metadata standard and the Warwick frame-word. The Dublin Core is a 15-Element Metadata Element Set proposed to facilitate fast and accurate information retrieval on the Internet [9]. Keywords are given different weights ranging from 0 to 1 (inclusive), depending on which element the keyword is in. Using sensitivity analysis, the weights are calibrated by the system designer, based on the importance and relevance of the keyword concerned.

To compute the page weight associating with different keywords, we initially select some keywords appearing in those pages. Each keyword is assigned a weight, ranging from 0 to 1, as mentioned previously. For example, the weight in the element Title should be higher than that in the element Description. The number of times a word appears in the page should also affect the page weight. Therefore, the page weight is computed as the sum of all products of keyword weight and the number of times that keyword appears in the page, using the following formula:

$$R_{(k_i)} = \sum_{i=1}^{n} (W_e * F_{k_i}) \tag{6}$$

where:

- R is the page weight based on the keyword k_i.
- k_i is the keyword appearing in the element e of the page $(1 \leq i \leq n)$.
- W_e is the pre-defined weight of the element e, determined by a system administrator via sensitivity analysis.
- F_{k_i} is the frequency that k_i appears in the element e of the page.

As in the research work in [5][7][8] shows that anchor text often provides more accurate descriptions of web pages than the page themselves. Anchors are texts around hyperlinks in a page p pointing to another page q. Therefore, we extend Dublin Core elements to anchors. We treat anchors as one of elements with a specified weight.

The equation (6) is further defined as:

$$R_{(k_i)} = \sum_{i=1}^{n}(W_e * F_{k_i} + W_a * F_{k_i}) \tag{7}$$

Where: W_a is the weight of anchors and F_{k_i} is the frequency of the keyword as an anchor text. In this case, if a keyword is an anchor text, then the product of W_a and the frequency will be summed up with the equation (6) as the weight of a page. $R_{(k_i)}$ is a relative page weight associating to keyword k_i. Therefore, a page has multiple weights in terms of different keywords:

$$R = \{ R_{k_1}, R_{k_2}, R_{k_3} \ldots R_{k_n} \} \tag{8}$$

Table 1 is an example of a page's weights. The second column of table 1 is the weight for different elements. From 3rd to 6th columns are the frequencies that those keywords in the first row appear in the corresponding elements. For instance, the figure of row No.2 and column No.3 is 1, which means the frequency of keywords "Information Filtering" in the element of title is 1 in this sample page. Likewise, the figure of row No.7 and column No.5 is 5, which means that keyword "Clustering" appears 5 times as an anchor text in pages pointing to this sample page.

Table 2. Page weights in terms of various keywords

Elements	Keyword Weight	Information filtering	Java tutorial	Clustering	Dublin Core
Title	0.95	1	1	0	1
Subject	0.82	0	1	0	1
Description	0.60	1	3	2	5
Content	0.3	8	3	6	4
Reference	0.50	9	6	2	5
Anchors	0.9	4	6	5	2
Page Weight	--	12.05	12.87	8.5	10.27

4 Re-ranking and Evaluation

4.1 Linear Combination

Previously, the page weight is obtained by analyzing hyperlink structure in section 2. Then, we calculate relative page weight in terms of different keywords based on K-elements defined in section 3. In this section, we describe the simple linear combination of page weight and K-elements as the re-ranking mechanism:

$$\text{Score} = \lambda P + (1-\lambda)R \tag{9}$$

In order to decide the value of λ, we tune λ from 0 to 1 and calculate the score in equation (9). Then rank the pages are ranked according to the score and users are asked to evaluate the results.

4.2 Evaluation

In this sub-section, we describe the setup of evaluation. The experiment is carried out in a simulating environment. Our web crawler crawls the web in a specified domain, where 326 pages and 3867 links are crawled. Each page's weight P_i is iteratively calculated based on equations (4) and (5). We select top 10 pages sorted by page weight P_i . An average of 10 keywords are specified for calculating R in equation (8). Subsequently, we have:

$$R = \{ R_{k_1}, R_{k_2}, R_{k_3} \ldots R_{k_{10}} \} \tag{10}$$

The final score of page weight is the combination of the equation (9) and (10):

$$Score_{(R)} = \lambda P + (1-\lambda) \{ R_{k_1}, R_{k_2}, R_{k_3} \ldots R_{k_{10}} \} \tag{11}$$

By tuning λ from 0 to 1, we obtain the Scores of page weight from equation (11). When users enter different keywords as query, the pages are re-ranked based on the final Score. These 126 users evaluate the rank shown in figure 2:

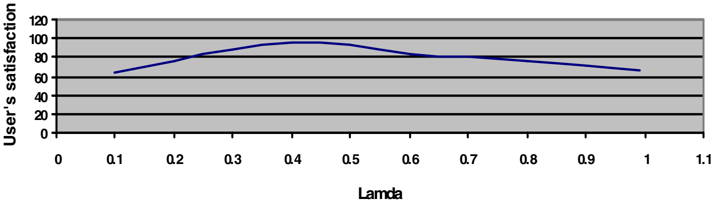


Fig. 2. User's evaluation of page rank

The x-axis is the value of λ and y-axis is the number of users who satisfy the page rank. As we can see from figure 2, the page ranks achieve the best performance when λ is set to around 0.4. From the above experiment, we could infer that it is not good enough to only consider hyperlink structure nor only rely on K-elements. However, when we use the linear combination of both the hyperlink structure and the K-elements, and set the coefficient λ to 0.4, the best performance of the search is achieved in this particular case.

5 Conclusions and Future Work

This research aims to rank web pages by the linear combination of hyperlink analysis and K-elements. Hyperlink analysis measures page weight by iteratively calculating the page weight based upon the weight of pages that it is connecting to and number of links on those pages. K-elements method computes the page weight in terms of different keywords by extending Dublin Core elements to anchors. Finally, the coefficient λ in the linear combination is determined by the user evaluation. The proposed method measures both of page importance and relevance.

In future work, we will be extending the K-elements method by the text surrounding anchors in addition to the anchor text itself. The future work also includes carrying out the evaluation with a great number of users who have different background and interests.

References

1. L.Page, S.Brin, R. Motwani, and T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web". Technical report, Stanford University Database Group, Jan. 1998. *http://dbpubs.stanford.edu/pub/1999-66*
2. Kleinberg, J., "Authoritative Sources in a Hyperlinked Environment", Journal of the ACM(JACM). Vol.46, No.5, 1999
3. R.Botafogo, E.Rivlin, B.Shneiderman, "Structural analysis of hypertext: Identifying hierarchies and useful metrics", ACM Transactions and Information Systems. Vol.10, No.2, 1992
4. Bharat, K.Henzinger, M.: "Improved Algorithms for Topic Distillation in a Hyperlinked Environment", Proceedings of ACM 21st International SIGIR'98, 1998, pp.104-111
5. Charkrabarti, S., Dom, B.: "Automatic Resource Compilation by Analysing Hyperlink Structure and Associated Text", Proc. The 7th International World Wide Web Conference, 1998, pp.389-401
6. X. Jiang et al., "Exploiting PageRank at Different Block Level", WISE 2004, LNCS 3306, 2004, pp.241-252
7. L.Page, S.Brin, "The Anatomy of a Large-Scale Hypertextual Web Search Engine".
8. Oliver A. McBryan. "GENVL and WWWW: Tools for Taming the Web". 1st International Conference on the World Wide Web. CERN, Geneva, Switzerland. May 25th-27th, 1994. http://www.cs.colorado.edu./home/mcbryan/mypapers/www94.ps
9. Dublin Core -- http://dublincore.org/documents/dces/
10. Angelaccio, M.; Buttarazzi, B., "Local searching the internet", Internet Computing, IEEE Vol.6, No.1, Jan.-Feb. 2002, pp:25 - 33
11. J, A Hartigan. Clustering Algorithms. WILEY Publication, 1975.
12. Salton, G. Automatic Text Processing. Reading, MA: Addison-Wesley Publishing, 1989.
13. N. Sankaran, "Speculation in the biomedical community abounds over likely candidates for nobel", The Scientist. 9 (19), Oct 1995. http://www.the-scientist.com/1995/10/02/1/1

14. J, Lai and B, Soh. “CRANAI: A New Search Model Reinforced by Combining a Ranking Algorithm with Author Inputs”, IEEE International Conference on e-Business Engineering, ICEBE 2005, Beijing, China, 340-345
15. Shardanand, U. & Maes, P. “Social information filtering: Algorithms for automating “word of mouth””. In *Proceedings of CHI'95 Conference on Human Factors in Computing Systems,* ACM Press, 1995, pp. 210-217
16. Basu, C., Hirsh, H. & Cohen, W. “Recommendations as classification: Using social and content-based information in recommendation”. In Proceedings of AAAI-98, 1998, American Association for Artificial Intelligence, 1998.
17. Weng, S. S & Liu, M. J. “Personalized product recommendation in E-Commerce” in Proceedings of the 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service, 2004, pp.413-420
18. Terveen, L., Hill, W., Amento, B., McDonald, D. & Creter, J. Phoaks: “A system for sharing recommendations”. Communications of the ACM, 40(3), 1997
19. Rucker, J. & Polanco, M.J. Sitesser: “Personalized navigation for the web”. Communications of the ACM, 40(3), 1997.
20. J, Lai and B, Soh. “Using Element And Document Profile For Information Clustering”, in Proceedings of the 2004 IEEE International Conference on e-Technology, e-Commerce and e-Service, 2004, pp. 503-506.
21. Baeza-Yates, R. and Ribeiro_neto, B., Modern Information Retrieval, Addison-Wesley, Reading, MA, 1999.
22. Anick, P. G. and Vaithyanathan, S., “Exploiting clustering and phrases for context-based information retrieval”, SIGIR Forum 31, 1, 1997, pp. 314-323.

Efficient Scheduling by Incorporating Bin Packing with Limited and Weighted Round Robin for Bluetooth[*]

Eung Ju Lee and Hee Yong Youn[**]

School of Information and Communication Engineering,
Sungkyunkwan University, Suwon, Korea
{oistrach, youn}@ece.skku.ac.kr

Abstract. In Bluetooth employing the conventional scheduling policies such as round robin, NULL packet is sent when the Master or Slave node does not have any data to send in its turn, and this causes a significant waste of resources. The Limited and Weighted Round Robin (LWRR) algorithm dynamically adjusts the resource allocation to each master-slave pair according to the queue status. In this paper we propose an improved LWRR (ILWRR) scheduling algorithm which effectively combines the LWRR and bin packing algorithm. Computer simulation reveals that slot utilization is increased up to about 50% compared to the round robin algorithm. The proposed ILWRR scheduling is effective for not only basic data transmission but also real-time multimedia data transmission.

Keywords: Bin Packing, Bluetooth, MAC, LWRR, slot scheduling.

1 Introduction

Recently, rapid prosperity of the wireless internet has spurred the research and development of personal area networks (PANs). The networks allow the electronic devices in close proximity to each other to form an infrastructure providing various services. Interconnection of consumer devices has been increasingly important as the use of hand-held devices such as PDA's and cellular phones has become quite popular. Bluetooth is a very attractive solution in the environment of handheld devices where the cables between portable and fixed electronic devices desired to be eliminated. It is a low power, short range wireless networking standard designed for local area voice and data communication [1]. Bluetooth employs a piconet structure that uses short range (10 meter) frequency hopping to improve the robustness of the link in the 2.4GHz band by avoiding interference from other devices such as microwave cooker. Piconet is an ad-hoc network, in which one of the devices acts as a master with the remaining ones acting as slaves.

Currently, Bluetooth uses time division duplex (TDD) scheme where communications for Master-to-Slave and Salve-to-Master strictly alternate. The master of the piconet is responsible for slot scheduling required for the pairwise communication. A

[*] This research was supported in part by the Ubiquitous Autonomic Computing and Network Project, 21st Century Frontier R&D Program in Korea and the Brain Korea 21 Project in 2005.

[**] Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 187 – 196, 2006.

frame consists of a set of time-division duplex slots with a Bluetooth packet occupying 1, 3, or 5 slots [1, 3, 4]. Bluetooth supports both voice and data traffic, and the nodes use Synchronous Connection Oriented (SCO) link communication or Asynchronous Connectionless Links (ACL). The most frequently used Bluetooth media access control (MAC) layer scheduling algorithm is PRR (Pure Round Robin) which is one of the conventional scheduling algorithms. In the RR scheduling each Master-Slave connection is alloted a pair of slots for the transmission of packets. Using the RR scheduling it is possible to provide each Master-Slave pair with a fair access to the channel. However, the simple RR scheduling is not efficient but causes resource waste [5]. This is because the Master can send packets to a Slave only in even numbered slots while the slave can send packets to the Master in odd numbered slots. This implies that the scheduling occurs in pairs of slots (i.e., the Master-Slave pair). Furthermore, since the task of scheduling is vested with the Master, there could be waste of slots and bandwidth if only the Master or Slave has data to send.

In order to solve the problem above, we propose a new scheduling algorithm effectively combining two algorithms - bin packing and Limited and Weighted Round Robin algorithm presented in [4]. The proposed algorithm called improved LWRR (ILWRR) algorithm significantly improves the utilization of slots and bandwidth in the Bluetooth MAC layer packet scheduling. Computer simulation reveals that slot utilization is increased up to about 50% compared to the PRR scheme. We also compare the throughput, delay, and the number of used slots of PRR, LWRR, and ILWRR.

The rest of the paper is organized as follows. Section 2 discusses the work related to Bluetooth technology and various slot scheduling algorithms. The proposed algorithm is introduced in Section 3, and the performance is evaluated and compared in Section 4. Section 5 concludes the paper.

2 Related Work

2.1 Bluetooth Technology

The protocols employed in Bluetooth are SDP, L2CAP, Link Manager, Baseband, and Bluetooth Radio. Bluetooth is a universal short range wireless communication system operating in the ISM band. The technology is based on frequency hopping using 79 carriers, 1 MHz spaced. Bluetooth channels use 1600hops/sec FH (Frequency Hop)/TDD scheme.

A network unit in Bluetooth is called a piconet. A piconet consists of at least two nodes: one master and up to seven slaves. The master defines pseudo-random frequency hopping sequence and transmission timing, and it uses centralized TDD scheduling as a MAC protocol. It also controls the channel strictly by polling the slaves and is always the first one transmitting a packet in one TDD cycle. Each slave may transmit a packet only after a successful reception from the master. Packet transmission from the Master starts in even numbered slots, while transmission from the slave starts in odd numbered slots. A Bluetooth packet can take 1, 3, or 5 slots. A slave may transmit a packet in the Slave-to-Master slot only if it is polled by the master in the preceding slot.

Here two types of links can be established. The first type, SCO, allows symmetric point-to-point service in which the master transmits packets in the reserved slots. The slave transmits packets in the succeeding slot. This scheme was designed to support real-time applications, especially voice. The second type, ACL link, is for data.

2.2 Polling Schemes for Bluetooth Piconets

In [4], several polling schemes are compared. In the pure round robin scheme a fixed order is defined and a single chance for transmission is given to each master-slave pair. The exhaustive round robin (ERR) algorithm also uses a fixed order but the master does not switch to the next slave until both the queues of the master and slave are empty. The main disadvantage of the ERR algorithm is that the channel can be occupied by the nodes generating higher traffic than the system capacity. A limited round robin (LRR) scheme limits the number of transmissions to solve this problem. A new scheme called LWRR (limited and weighted round robin) adopts weight on top of the limited round robin algorithm, which is dynamically changed according to the observed status of the queue [6]. Each slave is assigned a weight which is equal to the MP (Maximum Priority) at the beginning. Each time a slave is polled and no data is exchanged between the master and slave, the weight of the slave is reduced by 1. The lowest weight of a slave can attain is 1, in which case the slave has to wait a maximum of MP–1 cycles to get a chance to send packets. Anytime there is a data exchange between the slave and master, the weight of the slave is increased to the MP value. The rate of visit to a master-slave pair of a low weight is reduced to increase the bandwidth utilization [4].

2.3 Bin Packing

Bin packing (BP) algorithm is used to minimize the number of bins for packing some objects, which is an NP-hard problem [9]. There are two kinds of bin packing algorithm; on-line bin packing and off-line bin packing. Here we deal with only on-line bin packing since it is more suitable for Bluetooth environment. With on-line bin packing, each item must be placed in a bin before the next item is processed. With off-line algorithm, we have to wait until all the inputs arrive.

To solve the bin packing problem, a kind of greedy algorithm is applied. Greedy algorithms make decisions that look best at the moment. In other words they make decisions that are locally optimal in the hope that they will lead to globally optimal solutions. Unfortunately, decisions that look best at the moment are not always the best in the long run. These algorithms run quickly even though they do not necessarily produce optimal solutions. Since the on-line bin packing algorithm can effectively handle various size packets in Bluetooth environment, we adopt it. Now we examine the algorithm.

Assume that we are given some items of different sizes. The problem is to pack these items in the fewest number of bins. Figure 1 shows an example of optimal packing for seven items of sizes 0.2, 0.5, 0.4, 0.7, 0.1, 0.3, and 0.8. Note that at least three bins are required to pack the seven items since sum of the sizes of them is three.

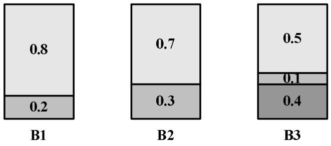


Fig. 1. An example of optimal packing for seven items

Now we show the NEXT FIT(NF) algorithm adapted in the proposed scheme. NEXT FIT is probably the simplest bin packing algorithm. For processing an item with NF, we check to see whether it can be put in the same bin as the previous item. If it does, it is placed there; otherwise, a new bin is created. This algorithm is incredibly simple to implement and runs in linear time. Figure 2 shows the result of packing for the example above. Notice that it needs a much larger number of bins than the optimal solution.

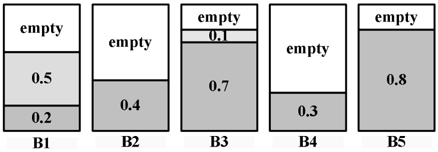


Fig. 2. An example of next fit packing for seven items

3 The Proposed Scheme

Now we present the proposed scheme that can efficiently allocate the resources in Bluetooth environment. The proposed scheduling algorithm consists of BP and LWRR algorithm. The scheduling algorithm for Bluetooth can be viewed as a online bin packing problem with limited amount of future knowledge. Note that there are 7 slaves in the piconet. BP can reduce the waste of slots by packing up to 5 slots. We assume that all the incoming data require 1, 3, or 5 slots equally likely. Figure 3 shows the structure of the scheduler operation.

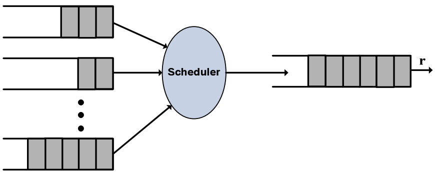


Fig. 3. The structure of the scheduler operation

w_i : Weight of flow$_\mathrm{i}$ (For RR, it is fixed. For LWRR, it is modified as the slot weight of the slave)

S_i : Amount of traffic service for flow$_\mathrm{i}$ between time t_1 to t_2

r : Service rate of the queue

r_i : Service rate of flow$_\mathrm{i}$

$$S_i(t_1,t_2)/S_j(t_1,t_2)=w_i/w_j \tag{1}$$

$$r_i/r \ =w_i/\sum_{k=t}^{t+n}w_k \tag{2}$$

In an ideal model each flow gets its fair share of bandwidth. The left hand side of Equation (1) is the ratio of service of *flow*$_i$ to *flow*$_j$ from t_1 to t_2, and the right hand side is the ratio of the weights of the two flows. For good fairness scheduling, we decide the service rate of a flow according to the weight.

One-dimensional bin packing is a classic problem with many practical applications related to the minimization of space or time. It is a hard problem for which many different heuristic solutions have been proposed. The goal is to pack a collection of objects into the minimum number of fixed-size "bins". Therefore, we use it with the LWRR algorithm to minimize the total number of used slots and increase the throughput as well.

Usually, all bins have the same capacity (C > 0). Obviously, placing an object in a separate bin is a feasible solution to the problem but not an optimal solution. The mathematical formulation for bin packing problem is represented as follows [7]. For *n* slots, the weight and capacity of *i* th slot are denoted as ps_i and c_i.

$$\min z(y)=\sum_{i=1}^{n}y_i \tag{3}$$

$$s.t. \quad \sum_{k=t}^{t+n}ps_jx_{ij}\le c_iy_i\ , \quad i\in N=\{1,2,3,...,n\} \tag{4}$$

$$\sum_{i=1}^{n}x_{ij}=1, \qquad j\in N \tag{5}$$

$$\textit{Where} \qquad y_i=\begin{cases}1, & \textit{if bin i is used}\\ 0, & \textit{otherwise}\end{cases} \tag{6}$$

$$x_{ij}=\begin{cases}1, & \textit{if object j is assigned to bin i}\\ 0, & \textit{otherwise}\end{cases} \tag{7}$$

For a given Problem I, it was shown that the solution with the NF algorithm, *NF(I)* satisfies the following bound

$$NF(I)\le 2z(I) \tag{8}$$

where *z(I)* denotes the value of optimal solution.

$$r_i/r \ =NF(w_i)/\sum_{k=t}^{t+n}w_k \tag{9}$$

w_i of Equation (2) is changed into NF(w_i) in Equation (9). It represents the proposed ILWRR algorithm showing a queue fair share of bandwidth and utilization according to the observed status of the queue. Figure 4 shows the pseudo code of the proposed algorithm, The LWRR scheduling algorithm displays better throughput than the RR algorithm but it is worse in terms of fairness for sending various size packets. Hence, we combine the NF bin packing algorithm with the LWRR algorithm to provide fairness.

Ideally, we can reduce the waste of slots as much as in Equation (8) by using the NF algorithm. In fact, however, the waste may be bigger because the packet distribution is not always uniform and capacity of the bins are not same. We will confirm the reduction of the waste of the slots with the proposed scheme in Section 4.

Nomenclature
MP: max slot priority
SW: slot weight of the slave
SK: the number of slots skipped for the slave

```
For each slave in the polling cycle:
01:  if(PacketS$_i$ <= size of Slot1) packetS$_i$ → Slot1
02:  while (If timer is not expired)
03: {
04:  Packets$_i$ += Packets$_i$
05:   if(size of Slot1 < PacketS$_i$ <= size of Slot3)
06:       Packets$_i$ → Slot 3;
07:    else  if(size of Slot3 < PacketS$_i$ <= size of
                slot5)
08:       PacketS$_i$ → Slot 5;
09: }
10:  else
11:   All Packets have not been packed
12: Poll the slave until SW time units expire or no
          data packet is exchanged
13:   if (no data packet is exchanged)
14:       SW=max((SW-SK),1);
15:    else SW=MP
```

Fig. 4. The proposed ILWRR algorithm in pseudo code

From line 01 to 09 in Figure 4, the NF algorithm is used to pack various size packets. From line 10 to 15, the LWRR scheduling algorithm is applied. With this composite algorithm, we can have better resource management than the RR and LWRR algorithm in practical condition.

We next show an example of actual slot scheduling with the RR and the proposed ILWRR scheduling algorithm when various size packets are input. Note that slot1 has two types of packets - DM1(0-17 byte long) and DH1(0-27). Slot3 has DM3(0-121) and DH3(0-183), and slot5 has DM5(0-224) and DH5(0-339), respectively, according to Bluetooth specification [1].

Example) A slave has six packets of the length of 15, 11, 61, 150, 122, and 125.

Sol) RR solution :
15(DH1) → assign slot1 and slot utilization α is 15/27 (56%).
11(DH1) → assign slot1 and α is 11/27 (41%).
61(DH3) → assign slot3 and α is 61/183 (33%).
50(DH3) → assign slot3 and α is 50/183 (27%).
122(DM5)→assign slot5 and α is 122/224 (54%).
125(DM5)→assign slot5 and α is 125/224 (56%).

Sol) LWRR solution :
15(DH1)→Do not assign slot1 but wait for succeeding packet.
11(DH1)→2 packets are packed and assign slot1. α is 26/27 (96%), and one slot has been used.
61(DH3)→Do not assign slot3 but wait for succeeding packet
50(DH3)→2 packets are packed and assign slot 3.
α is 111/183 (61%), and two slots have been used.
122(DM5)→Do not assign slot3 but wait for succeeding packet.
125(DM5)→The 122 byte and 125 byte cannot be packed because 122 + 125 = 247 > 224. Therefore 122 byte is assigned to slot5 and also 125 byte is assigned to slot5. α for the 122 byte is 122/224 (54%), and α for the 125 byte is 125/224 (56%).

4 Performance Evaluation

In this section we examine the performance of the proposed scheme that combines the LWRR algorithm with bin packing. The architecture of the proposed ILWRR scheduling is shown in Figure 5, which consists of several components such as Slave, Master, NF, and LWRR. The components are combined to decrease the waste of slots using the NF BP algorithm and increase the utilization of bandwidth using the LWRR algorithm. As mentioned earlier, the number of packets belonging to slot1, 3, and 5 are assumed to be equally likely. The packets are generated with Poisson distribution

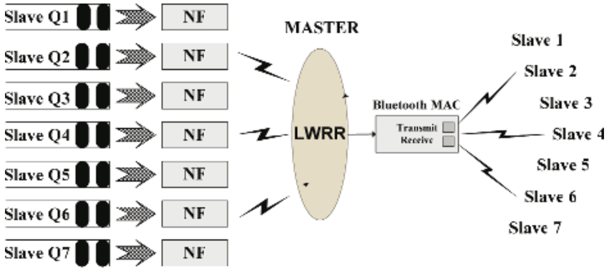


Fig. 5. The structure of the proposed scheme with the Bin Packing and LWRR algorithm

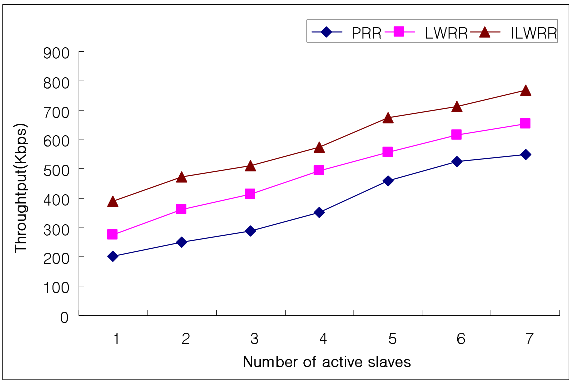


Fig. 6. The comparison of throughputs

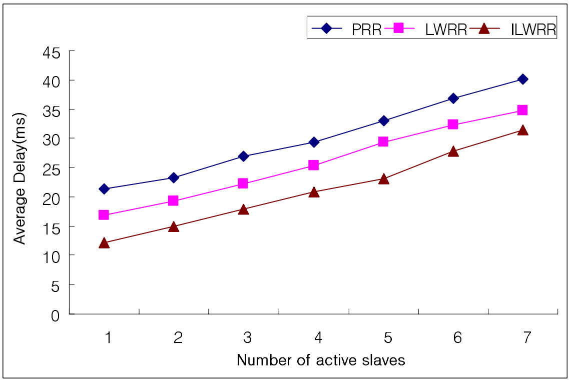


Fig. 7. The comparison of delays

with λ= 5. For each slot size, the lengths of the packets show uniform distribution. In other words, for slot1 packets, for example, the lengths of the payload sizes are evenly distributed between 0 and 27 bytes.

Figure 6 and 7 compares the throughput and average end-to-end delay of the RR, LWRR, and ILWRR scheduling algorithm in a piconet. The figures show that the proposed ILWRR algorithm consistently allows better performance than the other schemes. Figure 8 compares the total number of used slots with the three algorithms. Observe that the total number of used slots by ILWRR is minimum among the three. The average slot utilization is RR (53.45%), LWRR (71.51%), and ILWRR (80.51%), respectively. Note that the proposed ILWRR algorithm improves the utilization up to about 50%. Also, utilization of the proposed IDRR scheme is much more stable than the other two schemes. Recall that the NF bin packing is a very simple operation. Therefore, bin packing for a flow can be finished while the remaining six flows are handled in the LWRR-based Master. As a result, there exist no time overhead for

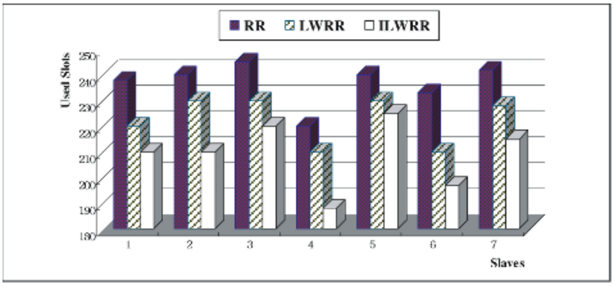


Fig. 8. The total number of used slots with RR, LWRR, ILWRR

implementing bin packing on top of the LWRR algorithm. The proposed ILWRR scheduling is thus effective for not only basic data transmission but also real-time multimedia data transmission.

5 Conclusion

In this paper we have proposed a new scheduling algorithm called ILWRR for Bluetooth by combining the BP and LWRR algorithm. Through computer simulation we have shown that the proposed algorithm performs significantly better than the Pure Round Robin algorithm in the Bluetooth environment. The PRR algorithm allows fairness to each slave node but causes the problem of wasted slot. The LWRR algorithm can increase the bandwidth utilization by dynamically adjusting the resource allocation to each master-slave pair according to the observed status of the queue. However, it still has wasted slot. The proposed scheduling algorithm, ILWRR, solves the problem by using the Next Fit bin packing algorithm along with the LWRR scheduling. More research is required to decide the optimized "bin(slots)" and "objects(packets)" parameter of the bin packing algorithm. The performance of the proposed scheduling algorithm for different distributions of packet sizes will also be investigated.

References

1. Bluetooth Special Interest Group (2005) Specification of the Bluetooth System, "http:www.bluetooth.com".
2. M. Shreedhar and G. Varghese (1996) Efficient Fair Queuing using Deficit Round Robin. Networking, IEEE/ACM Transactions on ,Volume: 4 Issue: 3 pp. 375–385.
3. M. Kalia, D. Bansal, R. Shorey (2000) Data Scheduling and SAR for Bluetooth MAC. IEEE VTC 2000-Spring Tokyo, pp. 716–720.
4. A. Capone, M. Geria, R. Kapoor (2001) Efficient Polling Schemes for Bluetooth Picocells. ICC, IEEE International Conference on Communication, pp. 1990–1994.
5. D. Yang, G. Nair, B. Sivaramakrishnan, H. Jayakumar, and A. Sen (2002) Round Robin with Look Ahead: A New Scheduling Algorithm for Bluetooth. International Conference on Parallel Processing Workshops (ICPPW02), pp. 45–50.

6. J.H. Kleinschmidt., M.E. Pellenz, and L.A.P. Lima Jr (2004) A Bluetooth Scheduling Algorithm using Channel State Information ICT'04 - 11th International Conference on Telecommunications, Aug 1-6, Fortaleza, Brazil.
7. J. Kang, S. Park (2002) Algorithms for The Variable Sized Bin Packing Problem. European Journal of Operational Research
8. A. Das, A. Ghose, A. Razdan, H. Saran, R. Shorey (2001) Enhancing Performance of Asynchronous Data Traffic over the Bluetooth Wireless Ad-hoc Network. IEEE INFOCOM 2001 , pp. 3211–3216.
9. B.S Baker and E.G Coffman (1981) A tight asymptotic bound for Next- Fit-Decreasing Bin Packing. in SIAM Journal on Alg. Disc. Meth, Volume: 2 pp. 147–152.
10. Bluehoc OpenBluetooth Simulator site http://www-124.ibm.com/ developerworks/ opensource/bluehoc
11. V. Sinha, D.R. Badu (2002) Class-based Packet Scheduling Policies for Bluetooth. National Conference on Communications: I.I.T. Bombay, pp. 25–27.
12. S. Keshav, (1999) An Engineering Approach to Computer Networking: ATM Networks, the Internet, and The Telephone Newtork, Addison-Wesley.
13. M.A. Weiss Florida International University (1994) Data Structures and Algorithm Analysis Second Edition,The Benjamin/Cummings Pub-lishing Company, Inc.
14. E.G. Coffman, Jr. and G.S. Lueker Approximation Algorithms for Extensible Bin Packing (2001) ACM/SIAM Symposium on Discrete Al-gorithms, pp. 586–588.
15. Bin-Packing linux based simulation site http://www.cs.arizona.edu/icon/oddsends/bpack/bpack.htm.

ECA Rule Component for Timely Collaboration of Web-Based Distributed Business Systems

DongWoo Lee[1] , Seonghoon Lee[2], and Yongjin Lee[3]

[1] Department of Computer Science, Woosong University,
17-2 Jayang-dong Dong-ku, Daejon 300-718 Korea
dwlee@woosong.ac.kr
[2] Department of Computer Science, Chonan University,
115 Anseo-Dong, Chonan, Choongnam 330-180 Korea
shlee@mail.chonan.ac.kr
[3] Department of Technology Education, Korea National University of Education,
Darak-Ri, Chungwon-Gun, Chungbuk 363-791, Korea
lyj@knue.ac.kr

Abstract. Timely collaboration among businesses is required to achieve their common business goals. In this paper an event-condition-action (ECA) rule component is proposed to support the timely collaboration of web-based distributed business systems. The proposed component provides high level rule programming and event-based immediate processing so that system administrators and programmers can easily maintain the timely collaboration independently to application logic. It uses HTTP protocol to be applied through firewalls and is implemented using basic trigger facilities of a commercial DBMS for practical purpose.

1 Introduction

In the WWW environment distributed business systems need to be coordinated and integrated for collaboration inter-organizations to achieve their common business goals. Especially emergency requests or urgent information among them should be processed in an immediate mode. For example, consider that a shopping mall becomes short of an item suddenly and requests a partner supplier to provide it. Then the supplier should provide the item quickly within a predefined time period. If, however, the item is out of stock in the supplier's warehouse and not able to be supplied within the predefined time period, it should be notified to the shopping mall promptly so that the shopping mall can try to find an alternate supplier to fill the item. Most current systems, however, due to the systems' security and autonomy, can not handle these requirements appropriately, but handle them in a batch processing mode or *ad hoc* manners [1].

In this paper an event-condition-action (ECA) rule component based on active database abstraction[2,3] is proposed to provide distributed business systems with flexible coordination and immediate processing in WWW environment. Since high level ECA rule programming is supported by the component, the collaboration among business systems and event-based immediate processing can be implemented independently to application logic.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 197 – 206, 2006.

The ECA rule component is designed and integrated into distributed business systems using HTTP protocol to be applied through firewalls. Thus the security and autonomy of distributed business systems of individual enterprise are assured. Since most of business systems have database systems to store persistent data and commercial DBMSs provide basic trigger functionalities of active capability, the component is designed and implemented using a commercial DBMS for practical purpose.

The rest of the paper is organized as follows. Section 2 discusses related work. Section 3 describes the need for timely collaboration among distributed business systems, the proposed mechanism to meet it, and requirements to provide the mechanism. In section 4, one of the requirements, elements of an ECA rule program and examples of ECA rule programs are presented. Section 5 discusses architecture of the ECA rule component and implementation and evaluation of a pilot system is described. Finally in section 6, we conclude with future work.

2 Related Work

The advent of the internet, WWW, and distributed computing technologies has been enabled business organizations to conduct business electronically. And a lot of researches on B2B E-Commerce have been carried on [4]. But the most of researches have been mainly focused on interoperability problems among distributed business systems. The issues of timely collaboration among business systems have not been addressed comprehensively.

There are many researches on exception handling issues on business processes [5, 6,7]. The exception is defined as deviation from the normal workflow, such as system errors or failures that interrupt normal processing of workflows. The exceptions are classified into basic failure on system level, application failure, expected exception on workflow level, and unexpected exception. Especially [5, 7] propose ECA rule based exception handling methods. However, these researches are focused on normal processing of workflows with the exceptions, i.e., fault-tolerant workflow processing. That is different from timely collaboration issues in this paper.

In summary, previous work addressed either general interoperability issues of businesses or exception handling of business processes' failures. Few of them comprehensively address timely collaboration issues among business systems in WWW.

3 Timely Collaboration Among Distributed Business Systems

3.1 The Need for Timely Collaboration

Consider a typical B2B E-Commerce shown in Fig.1 as a motivating example. All participants are connected by solid lines which denote internet and information flow. The dotted arrows denote material flow among participants. Each participant does business with each other by its own B2B and B2C systems. That is, a shopping mall takes customers' orders and provides services by B2C system, while it places orders to

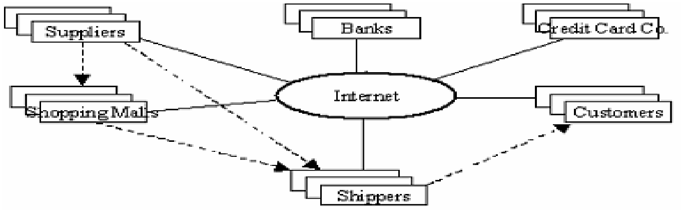


Fig. 1. Typical B2B E-Commerce

suppliers, requests delivery of items to shippers and money transfer to banks, inquires customers' credit to credit card company based on the agreement or contracts which were made with other participants. A supplier receives orders from shopping malls by its B2B system.

In Fig. 1 each participant has fire-wall for security and is connected each other via the fire-walls. In general, the fire-walls close most of ports except special ports such as HTTP 80. Therefore the systems of each organization should use HTTP protocol and fixed ports to interact with other organizations' systems[8].

In the above B2B E-Commerce there are two kinds of job processing modes in businesses. One is a batch-processing mode in which a business collects jobs and processes them at a time. The other is an immediate processing mode in which a business processes jobs promptly when they come in. In the former mode human or systems work efficiently while the processing time of each job becomes longer. In the latter mode human or systems should always wait for a job while each job is processed promptly. The choice of processing modes depends on the characteristics of jobs, policy of a business, and contracts between businesses.

Timely collaboration among business systems, i.e. immediate request - immediate cooperation, can be seen as exceptions out of business' normal collaborations. That is, emergency request or critical information among business systems should be transmitted to partners' systems promptly and processed by the systems in an immediate mode. They are not frequent, but once they occur they may require special treatment and affect customers' or businesses' profits in a large degree. Since businesses collaborate with each other by contract fulfillment, the cases for the timely collaboration can be classified as following in terms of service contracts:

1. unable to fulfill a normal contract service
2. need to modify or compensate a normal contract service
3. need to cancel a normal contract service
4. need a special service instead of a normal contract service

For the above cases, new contracts, which require timely collaboration, can be added into systems incrementally. Most current systems, however, due to the systems' security and autonomy, cannot handle these requirements appropriately, but handle them in a batch processing mode or *ad hoc* manners [1]. That is, they use login method by allowed users or Email. Or low-level *ad hoc* programs, which are coded into application logic, handle the cases. It causes software modularity problems.

3.2 Timely Collaboration by ECA Rule Paradigm

We derived the timely collaboration procedure among distributed business systems in B2B E-Commerce shown in Fig. 2, which consists of 4 phases:

1. Detection: a phase to detect that a business wants to make emergency requests for partner's cooperation or critical information occurs, which should be transmitted promptly.
2. Transmission: a phase to transmit the detected situation to a partner promptly.
3. Evaluation: a phase to evaluate collaboration constraints whether they should be processed in an immediate mode. There are two kinds of constraints, time constraints and resource constraints.
4. Processing: a phase to execute or process the requested job or the information in an immediate mode.

As shown above, in order to collaborate in an immediate mode, the situation for timely collaboration should be detected, notified or transmitted to each other, evaluated and recognized, and processed promptly. It shows that timely collaboration among distributed business systems is suitable application to ECA rule mechanism [2,3]. That is, the timely collaboration can be represented in ECA rules, such as occurrence of the situation for timely collaboration as event, collaboration constraints as condition, and the processing of the job as action. Then, an ECA rule component, which processes ECA rules, detects automatically the occurrences of the event and notifies the occurrence to partner's system. The component of the partner evaluates the condition. If the condition is satisfied then it executes the action promptly for collaboration. That is, the collaboration among business systems can be processed in an immediate mode without interference of application or users.

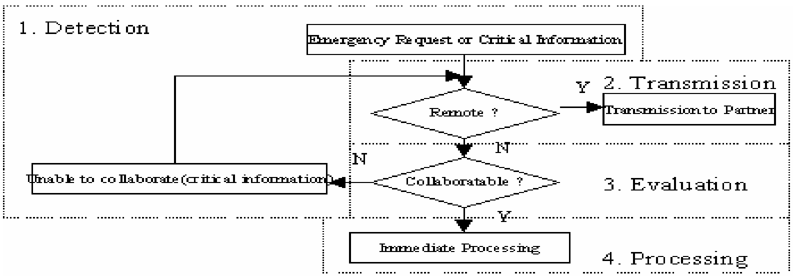


Fig. 2. Timely Collaboration Procedure among Distributed Business Systems

The timely collaboration among business systems in B2B E-Commerce can be supported by the two main elements; ECA rule programming facility and ECA rule component.

4 ECA Rule Programming for Timely Collaboration

In this research new ECA rule language is not developed. Instead an existent ECA rule language is extended at the minimum to express timely collaboration. We adopt ECAA(Event Condition Action Alternative Action) rule language which is one of ECA rule languages[2,3]. Fig. 3 depicts ECA rule structure. Alternative Action part is for flexible representation of timely collaboration. Even though a business makes a timely collaboration request, its partner may not be able to cooperate because of the collaboration constraints. In this case the partner should notify the inability for cooperation to requester as an alternative action. Therefore it is optional.

```
Rule rule-name;
   Event event-expression;
   Condition condition-expression;
   Action Begin action-block End
   [Alternative Action Begin alternative-action-block End]
```

Fig. 3. ECA Rule Structure

The rule-name does not have a major role in its execution since it is triggered by an event. Rule names are used mainly for management purpose.

Event: A rule is triggered by detection of occurrence of an event described in event-expression. The need for the timely collaboration among business systems in B2B E-Commerce is represented as an event. In this paper the events are classified into local and remote in terms of occurrence and subscription. If a local event occurs and is subscribed by a remote system, it is transmitted to the system. This information is registered into Event-Schema-Table of Event-Manager in an ECA rule component when it is defined. In terms of contents, the events are furthermore classified as request event and notification event as in [Table 1]. Since events for timely collaboration are application-related, the wrapper codes are required to generate them and transfer to Event-Manager. The syntax of the event-expression is

event-expression ::= event-name(type1 par1, type2 par2, ...);

Table 1. Events and Actions for Timely Collaboration

Request Events & Notification Events	Immediate Collaborative Action
Notify-Able-Service	No-Action
Notify-Unable-Service	Find-Alternate-Service
Request-Modify-Service	Modify-Service
Request-Cancel-Service	Cancel-Service
Request-Special-Service	Special-Service

Condition: Once an event has been detected, a condition part is evaluated. For timely collaboration, collaboration constraints should be checked. Even though a business system requests timely collaboration to its partner system, the partner may not be able to cooperate because of collaboration constraints. There are two kinds of constraints. One is time constraints that the partner system should provide requested service within a predefined time period. Another is resource constraints that the partner should have resources such as man power, required system, or items to fulfill the requested service.

Action: If the condition is satisfied, the action block is invoked. An action block consists of a set of actions or call statements which execute the requested service for timely collaboration. [Table 1] shows kinds of requested events and notification events and corresponding action types. Some actions may generate events again.

Alternative Action: If the condition is not satisfied, which means the partner system can not cooperate immediately, the alternative action block is invoked. It is used to notify the inability of collaboration to the requester or remedy it.

Example of ECA rule programs: Consider the previous example that a shopping mall becomes short of an item suddenly and requests a partner supplier to provide it. Then the supplier should provide the item quickly within a predefined time period. If, however, the item is out of stock in the supplier's warehouse and not able to be supplied within the time period, it should be notified to the shopping mall promptly so that the shopping mall can try to find an alternate supplier to fill the item.

This example shows that a shopping mall and a supplier should collaborate in an immediate mode. It can be implemented by the following two rules ;

```
Rule Find-Alternate-Service /* rule on a Shopping Mall */
         Event unable-special-supply(string supplier, string item-1, integer n);
         Condition true;
         Action Begin find-alternate-service(string item-1, integer n)
              End

Rule Request-Special-Service /* rule on a Supplier */
         Event request-special-supply(string requester, string item-1, integer n);
         Condition no. of item-1 > n;
         Action Begin special-order-processing(string requester, string item-2) End
         Alternative Action Begin raise-event('notify-unable-special-supply') End
```

5 ECA Rule Component

5.1 Architecture of ECA Rule Component

The timely collaboration written in ECA rules is processed by an ECA rule component. The architecture of the component consists of five modules, Communication-Manager, Event-Manager, Rule-Manager, Event-Rule-Interface, and Actions/Applications. The overall architecture is shown in Fig. 4. In the following we describe each module.

Communication-Manager: Using HTTP protocol the Communication-Manager sends and receives event messages for timely collaboration in the form of XML with Communication-Manager of partner's ECA rule component. It is implemented in Java servlet[9] of a Web server and contains two roles. Firstly it receives XML messages through a Web server, extracts events, and transfer to local Event-Manager. Secondly it receives event from local Event-Manager, transforms into XML format, and using HTTP post command transmits to the Communication-Manager of partner's system[10]. In order to process XML messages Xerces2 Java parser is used. Send_Event and Raise_Event take the two roles respectively.

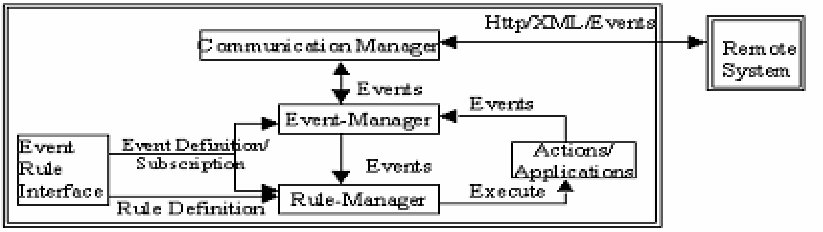


Fig. 4. Overall Architecture of ECA Rule Component

Event-Manager: It manages schema definition of events and their subscription, identifies whether subscription of an event is local or remote, and transfer the identified event to corresponding Rule-Manager which subscribes it. It is implemented in Java servlet of a Web server. Events are defined and registered for subscription by system administrators or programmers via Event-Rule-Interface. Then the definition and subscription are recorded into Event-Schema-Table by Event- Manager. At run time the Event-Schema-Table is referenced by Event-Manager to check whether an event is local or remote. The Event-Schema-Table consists of four tables to store all information related to events. Four tables are;

event-schema=(event-name, no-of-parameters)
publisher-schema=(event-name, publisher)
subscriber-schema=(event-name, subscribers)
parameter-schema=(event-name, para-name, type, position)

Since Event-Schema-Tables are stored in database, Event-Manager uses JDBC to connect database for storing and retrieving events.

Rule-Manager: The Rule-Manager evaluates and executes ECA rules. It is implemented with basic trigger facilities of the underlined DBMS and contains Event-Instance-Tables, a Rule-Table, and triggers on the Event-Instance-Tables. An ECA rule for timely collaboration is defined via Event-Rule-Interface and recorded into the Rule-Table. When an event is defined, its Instance-Table is created together.

A rule for timely collaboration is written using triggers on the Instance-Table. For example, the rule 'find-alternate-supplier' can be written as a trigger on the Instance-Table 'unable-special-supply-instance-table' in the syntax of Oracle 9i[11];

```
create trigger find-alternate-supplier after insert on
                                        unable-special-supply-instance-table
begin
        if condition is true
        then call find-alternate-supplier();
end;
```

Event-Rule-Interface: The Event-Rule-Interface is an interface for system administrator or programmers to define events and rules as well as to manage them, *i.e.*, search, delete, and update. It utilizes facilities of Event-Manager and Rule-Manager, that is, Event-Table-Editor and Rule-Table-Trigger-Editor, respectively.

Actions/Applications: Actions are the treatments of requested services. Events are generated by applications. And some actions may generate events again. The actions/applications are internal if written in underlined DBMS's API or external if not. Since events for timely collaboration are related to applications, wrapper codes to generate their occurrences and function calls to notify to Event-Manager are needed. Since the action part of an ECA rule is to treat timely collaboration, it consists of procedure or call statement for stored procedure, which provides the collaboration.

5.2 Interactions Among Component Modules

In order to explain how the ECA rule component processes the ECA rules for timely collaboration, the interaction procedure among modules is described at build time and run time in the following, respectively.

Build Time

1. System administrators or programmers define events for timely collaboration via Event-Rule-Interface.
2. The definition of events are recorded into Event-Schema-Table by Event-Manager. System programmers write wrapper codes for occurrences of the events.
3. The definition of an event is transferred to Rule-Manager and its Instance Table is created.
4. Subscription of the event is registered via Event-Rule-Interface by the partner's programmer. The Subscription is stored into Event-Schema-Table by Event-Manager.
5. System administrators or programmers define a ECA rule on the event via the interface. The definition of a rule is recorded into Rule-Table by Rule-Manager.
6. System programmer writes triggers on the related Event-Instance-Table of the defined ECA rule and codes for the action part.

Run Time

1. A local event occurrence generated by a local application or action is notified, or a remote event is transmitted from a remote system via Communication- Manager.
2. Event-Manager identifies its subscription from Event-Schema-Table. If its subscription is local, then it is transferred to the local Rule-Manager. Otherwise, that is, remote subscription, it is transferred to Communication-Manager to transmit to a corresponding remote system.
3. Rule-Manager inserts the event instance into its Instance-Table and invokes a trigger to execute corresponding rule.
4. The trigger executes corresponding action for timely collaboration.

5.3 Implementation and Evaluation of a Pilot System

To validate the timely collaboration mechanism and applicability of an ECA rule component, a pilot system has been implemented and applied to a typical B2B E-Commerce scenario. During design and implementation of the system, we considered practicability, interoperability with database, and platform independence. Therefore Java was chosen as an implementation language. Commercial DBMS Oracle 9i, Apache Tomcat Web server, Xerces2 Java parser were chosen.

Instead of implementing all features of an ECA rule component from scratch we utilized basic trigger facilities of a commercial DBMS. Major functions of the component are detection of event, evaluation of condition, and execution of action. In this paper, as described in the previous section 5.1 the condition evaluation and the action execution parts are designed and implemented by triggers of a underlined DBMS. That is, Event-Instance-Table for each event is created in DB and trigger codes on this table are written. The condition evaluation and action parts are written in trigger body. Thus, when an event instance is inserted into its Instance-Table, it invokes the related trigger that evaluates the condition and executes action part in the body.

For the application scenario, an internet shopping mall with its suppliers and shippers is used as shown in Fig. 1. Consider previous shopping mall. It decided to provide flexible service that customers are allowed to modify(add or delete items) or cancel their orders, and modify the delivery address within a predefined time period. In order to provide these services the shopping mall should collaborate with suppliers and shippers. Furthermore, since these collaboration affect the normal jobs processed, being processed, and going to be processed, they should be processed in an immediate mode. The events and rules for timely collaboration among shopping mall, suppliers, and shippers for the scenario are shown in Fig. 5.

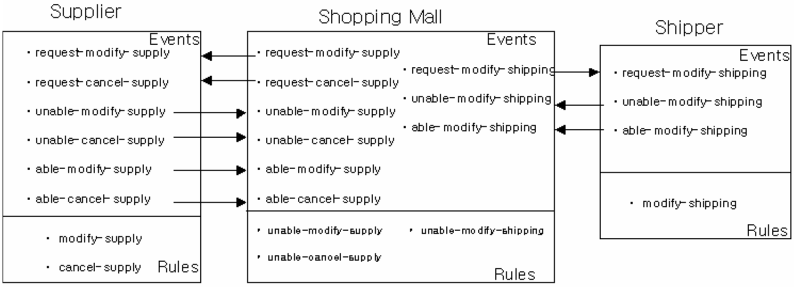


Fig. 5. Events and Rules for Timely Collaboration

6 Conclusion

In this paper an ECA rule component based on active database abstraction is proposed to support timely collaboration among distributed business systems in WWW environment. Since high level ECA rule programming is supported by the component, the collaboration among business systems and event-based immediate processing can be implemented independently to application logic. Thus, system administrators and programmers can easily program and maintain timely collaboration among business systems in WWW.

The ECA rule component is designed and integrated into business systems using HTTP protocol to be applied through firewalls. The security and autonomy of systems in individual enterprise are assured. Since most of business systems have commercial database systems to store persistent data and commercial DBMSs provide basic trigger functionalities of active capability, the ECA rule component is implemented to be practical using a commercial DBMS, Oracle. Even though the syntaxes of triggers of commercial DBMSs are different from each other, since the semantics of the triggers are similar, the proposed component can be implemented in other commercial DBMSs such as DB2 and MS SQL server.

In order to extend our research, future work includes support for various phases in business systems' life cycle[12] and extension of collaboration concept to support client oriented ECA rules. In addition, research on security of event transmission and action execution is needed, since notification of an event leads action execution directly. It may affects applications and database. Thus appropriate security method is required.

References

1. Nobuyuki Kanaya, et. al., "Distributed Workflow Management Systems for Electronic Commerce", proceedings of 4th International Enterprise Distributed Object Computing Conference(EDOC'00), IEEE 2000.
2. Norman W. Paton and Oscar Diaz, "Active Database Systems", Computing Surveys, ACM, 1999.
3. J. Widom and S. Ceri, *Active database Systems, Triggers and Rules for Advanced Database Processing*, Morgan Kaufmann, 1996.
4. Brahim Medjahed, et al., "Business-to-business interactions: issues and enabling technologies", *VLDB Journal*, Springer-Verlag, April, 2003. pp.59-85.
5. Fabio Casati, S. Ceri, S. Paraboschi, and G. Pozzi, "Specification and Implementation of Exceptions in Workflow Management Systems", ACM Tr. on Database Systems, Vol. 24, No. 3, September 1999, pp.405-451.
6. 6 Zongwei Luo, Amit Sheth, Krys Kochut, and Budak Arpinar, "Exception Handling for Conflict Resolution in Cross-Organizational Workflows", Technical Report, LSDIS Lab, Computer Science, University of Georgia, April 10, 2002.
7. J. Meng, Stanley Y.W. Su, herman Lam and A. Helal, "Achieving Dynamic Inter-Organizational Workflow management by Integrating Business processes, Events and Rules", IEEE HICSS-35'02, 2002.
8. Aaron Skonnard "SOAP:The Simple Object Access Protocol", http://www.microsoft.com/mind/0100/soap/soap.asp, Microsoft Internet Developer, January 2000.
9. Danny Coward, "Java Servlet Specification version 2.3", Sun Java Technology, Release 8/31/01.
10. R. Fielding, et al., "Hypertext Transfer Protocol -- HTTP/1.1", W3C, 1999.
11. Oracle9i SQL Reference Release 2 (9.2) Part Number A96540-02, October 2002.
12. David Trastour, Claudio Bartolini, Chris Preist, "Semantic Web Support for the Business-to-Business E-Commerce Lifecycle", proceedings of www2002, 2002.

Dynamic Approach for Integrating Web Data Warehouses

D. Xuan Le, J. Wenny Rahayu, and Eric Pardede

Department of Computer Science and Computer Eng.,
La Trobe University Bundoora VIC 3083, Australia
{dx1le, wenny, ekpardede,}@cs.latrobe.edu.au

Abstract. This paper proposes a dynamic integration approach for a web data warehouse system. It starts with a conceptual design begins with specified requirements that allow the need for creating a logical integrated web data warehouse model. An object-oriented concept is utilized at the logical design level to fully capture and represent the semantics of underlying data sources and user's requirements in a more flexible and meaningful manner. We also show the benefits of our proposed integrated web data warehouse solution by presenting a set of complex queries to access the integrated data for computing complex analytical results.

1 Introduction

Integration of distributed data is aggressively becoming more of a concern as more business data appear on the World Wide Web. The business information is stored in more than one places, the data from these various sources are needed to build an analysis tool to support business and management decisions in order to keep an enterprise's competitive edge. Data integration is a very challenging topic and very often the data integration in specific areas is misinterpreted.

The challenge behind the integration of web data and non-web data is that data from different resources are almost impossible or not related to business subject or even some data appeared to be very inconsistent in their representation. Therefore, in order to represent the selected data in different resources to correspond to the same subject, semantics and structure of data must be carefully considered. For example, XML/HTML documents provide many possibilities semantics to model a real world problem. For a same problem, data are split in XML, relational and object semantics. The question is how are these semantics are captured and formed a uniform data warehouse structure. These factors motivate us to define a uniform integrated web data warehouse structure that allows distributed data to be mapped and integrated accordingly.

Our proposed solution on integrated web data warehousing includes the conceptual design level, logical design level and the benefits of the proposed approach that make it outperform the existing proposed techniques when complex queries provide different views of integrated data. Furthermore, our proposed approach is a completion from conceptual design to implementation level. The design strategy and methodology are effortlessly understood by all levels, and highly assist the system designer to handle more data models and provide the flexibility of maintenance in the future.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 207 – 216, 2006.

This paper is organized as follows. Section 2 briefly describes the existing works. Section 3 is a preliminary on data integration in data warehouse. It also gives a brief description of the framework and integration architecture on different data models. Section 4 is the definition of the conceptual design level, while section 5 is the schema creation including a complex query specification. Section 6 is the conclusion of the paper, discussion of the system benefits and proposed extension of this paper.

2 Related Work

Proposed techniques in [2, 3, 4] such as view mechanism, based classes and the mapping of local schema to a global schema have explicitly adopted object oriented (OO) data modeling concept to allow a creation of an object model to deal with a design problem and data mapping issues. These proposed techniques have focus on the handling of complex and object data types in a data warehouse system. The lacking of features and data representations on an OO data warehouse model extended from an existing relational data warehouse model in [9] shows the insufficient semantics of data representation.

XML structure is a popular standard technique of storing and exchanging web data; XML structure has the capability to comply with OO representations. Because of its dominant storage and exchange for web data that has attracted more researchers to investigate the design and building techniques of XML Data warehouse (XDW). A logical designing technique for XML data source integrated in a data warehouse using relational data model could be found in [5]. While a web-based data warehouse is constructed based on XML data to solve evident defects in [7], a dynamic data warehouse that allowed an integration of XML data in [11] was proposed to support the evaluation and change control over the integrated data.

The XDW refers to a proposed XML document data warehouse in [10] in which OO concept is utilized to assist with the conceptual design. The design of XDW is driven due to the concern of the fast expansion in large amounts of sources in the XML documents. An OLAP specifying on XML data and relational data in [6] is concerned with DTD structure for XML data and assume relational data can be easily done by using a direct mapping technique. The authors explicitly adopt UML tools to create the integrated data model. Their main focus is to deal with XML data and therefore uses XML as a design based data structure. Relational data sources is then observed and directly mapped onto the defined model.

3 Data Integration in Data Warehouse: A Preliminary

Our web data warehouse integration approach begins with a conceptually predefined common integrated web data warehouse model that has a multidimensional structure. This common data warehouse model is specified, based on user and business requirements and the underlying sources combined; the conceptual design utilizes OO features. The definition of the model focuses on the ability to integrate different data models in the data sources and fully capture the semantics of data. The specified model has a structure of *Fact* revolved by a set of hierarchical *Dimensions*.

Our proposed integrated data warehouse model is very much similar to the Star model. However, the advantage of our model is that the dimensions are explicitly organized in hierarchical structures with the assistance of OO features to provide a full capture of data semantics and representations for business requirements.

In the completion of a conceptual design for an integrated web data warehouse model, data sources are then absorbed and modeled based on the specified modeling features in which a logical model is created. A set of complex queries is also proposed to access the integrated data to compute analyzed results.

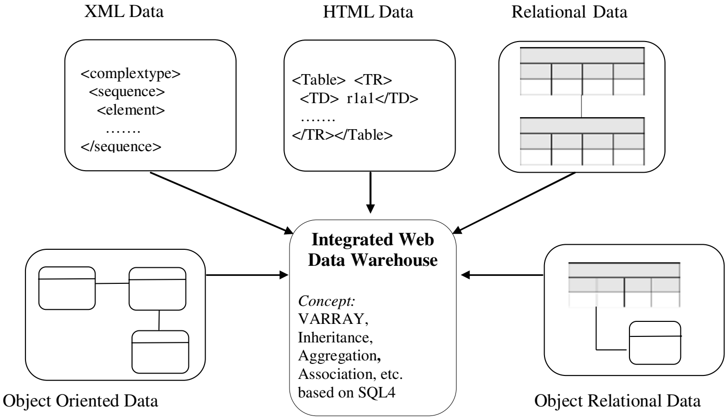


Fig. 1. Integrated Web Data Warehouse Model Overview

Fig. 1 shows an overview of our proposed web integrated data warehouse architecture. Our goal is to present the technique to integrate Relational, XML and HTML, Object Relational (OR) and OO data.

3.1 Integrating Dimension Data

We explicitly use OO features to deal with dimensional hierarchical structures. Thus, each dimension may hierarchically store information to represent different business requirement in which an aggregation, inheritance or association relationship is used. Each type of relationship is selected according to the specified requirements and knowledge of structure in the underlying sources. The cardinality of –to-one or –to-many shows the interrelationship between the data levels wherever it is appropriate. It is not necessary to specify relationships between classes in a single hierarchy, especially when it is engaged with inheritance or collection.

3.2 Integrating Summarized Fact Data

We propose an association relationship of one-to many between Fact and each dimension. Fact stores the pre-aggregated measures and a set of referenced keys that refers to the dimensions. The pre-aggregated measure is a summarized data value that has references to dimensions. This value is computed using reference dimension data that are stored in different data source models. Since the proposed dimensions in our integrated model utilized the hierarchical concept, the pre-aggregated measures are derived as at lowest level as possible in the dimension. This allows the users flexibility to compute powerful analyses on the specific OLAP queries.

3.3 Integrating Different Data Models

In our proposed model, we aim to integrate two main data type, relational data and web data such as HTML/XML documents. Relational data is still necessary since an enterprise may not be fully equipped with web-based systems. Some transactional data are still stored in the operational system. We consider integrating the relational data directly onto the common web data integrated model so that we can reuse the existing relationship between the data as much as possible. If necessary we map from relational data to an OO structure in which we can gain more semantic data representation by using OO modeling features.

When dealing with HTML/XML documents, we use a powerful formalism of XML schema to describe the document structure and user requirements. The specified XML Schema is also used to validate data that are captured or given in XML structure. XML schema, which is formally recommended by W3C, can also accommodate the OO modeling concepts. We propose to translate from HTML data to a XML schema [1, 8]. We assume that the data present on the web for a particular interest subject is described in a table that use <Table>, row <TR> and column <TD>, <TH>, <Colspan> or <Rowspan> tags. The mapping is carried from each of these tags to each of the structured name in the desired XML schema.

4 Defining Conceptual Integrated Web Data Warehouse Model

As a motivating example for this paper, we illustrate the proposed approach based on the requirements to build an enrolment university data warehouse system. Information about the enrolments is stored in relational and web forms. This is due to the fact that individual faculty has its own system and they are not currently linked.

One faculty may have its own web-based system while the others for some reasons may just have a relational database system to handle the enrolment of students. It is the goal of the university to construct a data warehouse system in order to analyze student enrolments, favorable interested areas/subjects/degrees and also the trend of enrolments in different years including month and semesters.

The university is also interested in the analysis of degree enrolments for particular area. For example, for masters degree there maybe more students enrolled in course work than research but exceptionally a university has a strong constraint in providing both research and coursework. Thus, it is interesting to see a pattern among these entities. A faculty may be formed by one or more schools and a certain number of degrees belongs to a particular school.

4.1 Translating HTML Data into XML Structure

We adopt the mapping tool and technique that proposed in [1, 8] to map from HMTL data to XML data so that attributes can be identified. HTML data are translated to XML schema using very basic and obvious mapping steps.

Let the content of table XYZ is a set of rows <TR> and each row contains a set of column <TD> then XYZ is mapped to an XML Schema structure; <TR> is mapped to the <xsd:Sequence>; <TD> is mapped to the <xsd:elment> wihin the sequence.

4.2 Conceptually Specifying Integrated Web Data Warehouse Model

Conceptually, starting with the assumptions of the user specified requirements and information related to underlying sources in Relational and XML. A *conceptual defined sequence* is defined to assist with the process of creating the model:

1. Simplifying the requirements. Structures of underlying data sources can also be simplified if possible.
2. Defining dimensions in two steps (a) specifying n classes where n ≥ 1; (b) classifying hierarchy: suppose two classes A and B in a dimension, the relationship between A and B can either be i, ii or iv:
   i. Aggregation: using existence dependent and existence independent to describe the relationship between the whole-part classes.
   ii. Inheritance: categories subtypes and super-types
   iii. Collection: handles multi values in an attribute
   iv. Association: using a -to-one; -to-many to describe the association between classes
3. Defining Fact in two steps (a) a simple single Fact; (b) considering the cardinality between Fact and Dimensions, which is only a one-to-many relationship.

Extracting from the *case study* '…The university is also interested in the analysis of degree enrolments for particular type, for example, for masters degree there maybe more students enroll in course work than research but exceptionally a university has a strong constraint in providing both research and coursework…', using the *conceptual defined sequence,* we have the followings:

1. Simplified requirements {Categorized degree by research/coursework, performed by enrolment}
2. Dimension {Degree}
   a. Classes {Degree, Research, Coursework}
   b. Hierarchy{Generalization}

Based on the above rules, a conceptual degree dimension is derived as shown in Fig. 2. below.

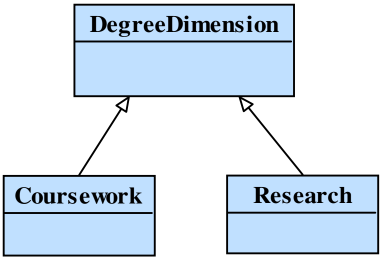


Fig. 2. A Conceptual Degree Dimension

5 Defining A Logical Integrated Web Data Warehouse Model

In this section, we propose a logical design of the integrated data warehouse which is based on the case study in previous section. A complex query is also specified based on our developed prototype.

5.1 Definitions for Integrating Logical Dimensions and Fact Classes

Defining integrated logical Dimensions class, using conceptual dimensions in previous section, consists of three steps: (a) specifying attributes and any special kind of attributes that support the design or derivation; (b) specifying integrated attributes to individual classes and specifying an object id, which is also known as a primary key (PK) in each base class; and (c) data types may also be considered but not necessary shown in the design model.

Dimension (class name {attributes, hierarchy, cardinalities})

Defining integrated logical Fact class, using conceptual dimensions in previous section, consists of three steps: (a) specifying the attributes including referenced keys to dimension; (b) deriving aggregated textual measures; and (c) specifying a link between Fact and Dimensions with 1..* to 1.

Fact (classname {attributes, hierarchy, cardinalities})

Fig. 3. shows an OO integrated web data warehouse model. Subject dimension is modeled using a collection type to define a *prerequisite* attribute as a VARRAY type that allows to multiple values.

While the FacultyDimension has a non-sharable existence dependent relationship between the faculty and school classes, the TimeDimension has a sharable existence independent relationship between the time and semester classes.

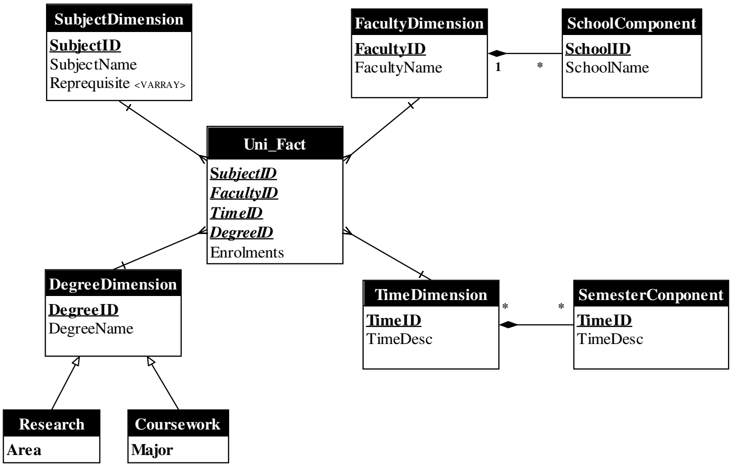
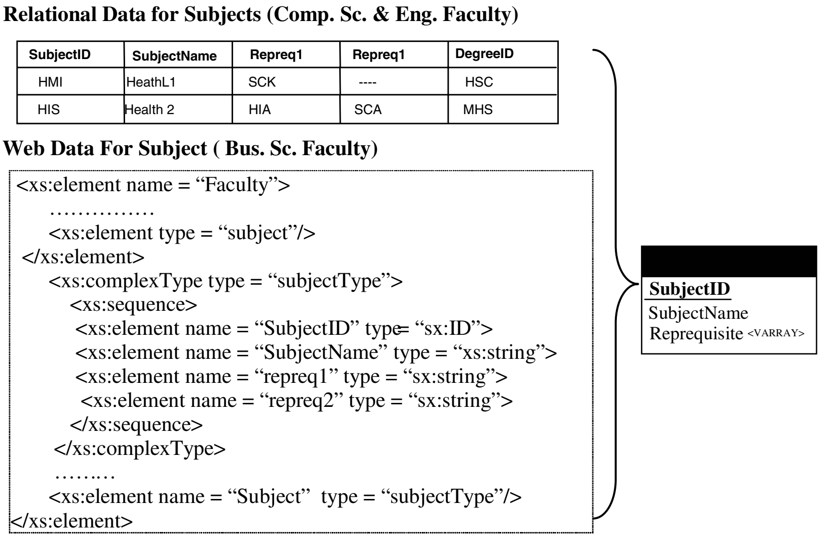


Fig. 3. University Integrated Web Data Warehouse Model

5.2 Defining Integrated Logical Dimensions and Fact Classes for Case Study

In this section we use the proposed method for the case study. We start by specifying flat level dimension classes. Fig. 3. shows the relational data and web data for the subjects. An XML Schema is specified based on the provided XML/ HTML data. Following the *sequence* in section 4.2 and definition in 5.1, we specify information for the *SubjectDimension* such as *Integrated attributes* {SubjectID, Subjectname,



Relational Data for Subjects (Comp. Sc. & Eng. Faculty)

SubjectID	SubjectName	Repreq1	Repreq1	DegreeID
HMI	HeathL1	SCK	----	HSC
HIS	Health 2	HIA	SCA	MHS

Web Data For Subject (Bus. Sc. Faculty)

```
<xs:element name = “Faculty”>
     ……………
     <xs:element type = “subject”/>
 </xs:element>
     <xs:complexType type = “subjectType”>
        <xs:sequence>
         <xs:element name = “SubjectID” type= “sx:ID”>
         <xs:element name = “SubjectName” type = “xs:string”>
         <xs:element name = “repreq1” type = “sx:string”>
          <xs:element name = “repreq2” type = “sx:string”>
        </xs:sequence>
      </xs:complexType>
      ……..
     <xs:element name = “Subject”  type = “subjectType”/>
</xs:element>
```

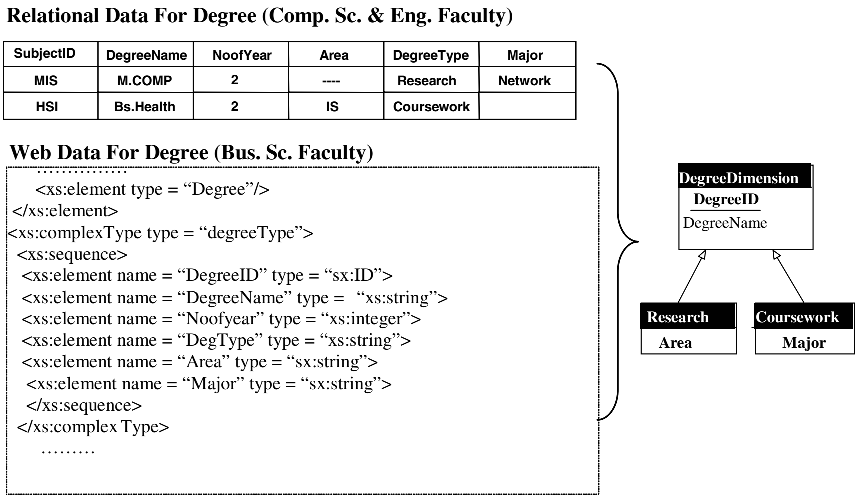
Fig. 4. Derived Subject Dimension from given XML and Relation data

Prereq1, Prereq2} are simple data types; group attribute {prerequisite (PreReq1, PreReq2)}; *Link type* {None}; PK {SubjectID}. The integrated dimension is defined as follows.

Dimension (Subjec{Integrated Attributes, Collection Type, Nil})

In Fig. 4, we use a collection type to store multi values of the Prerequisite attribute which provides the flexibility to store a series of data entries that are jointly associated to a corresponding row. We implicitly apply the VARRAY because it is sufficient to store multi values that have a simple data type and the number of elements is known in advance.

Next step is specifying hierarchical dimension Classes. Fig. 5. provides relational data about degrees stored in Degree Table and XML/HTML data stored in an XML/HTML document in which an XML Schema is specified.



Relational Data For Degree (Comp. Sc. & Eng. Faculty)

SubjectID	DegreeName	NoofYear	Area	DegreeType	Major
MIS	M.COMP	2	----	Research	Network
HSI	Bs.Health	2	IS	Coursework	

Web Data For Degree (Bus. Sc. Faculty)

```
   ……………
   <xs:element type = "Degree"/>
</xs:element>
<xs:complexType type = "degreeType">
 <xs:sequence>
  <xs:element name = "DegreeID" type = "sx:ID">
  <xs:element name = "DegreeName" type =  "xs:string">
  <xs:element name = "Noofyear" type = "xs:integer">
  <xs:element name = "DegType" type = "xs:string">
  <xs:element name = "Area" type = "sx:string">
   <xs:element name = "Major" type = "sx:string">
  </xs:sequence>
 </xs:complex Type>
    ………
```

Fig. 5. Derived Degree Dimension from given XML and Relation data

Following the *sequence* in section 4.2 and definition in 5.1, we specify the important information that supports the derivation of the *DegreeDimension. Integrated attributes* {DegreeID, Degreename, degType, Area, Major} are simple data types. Type attribute is degType, which is used to add additional information about a degree that is either a *Research* type *or Coursework* type. *Link type* {Inheritance hierarchy} is based on the present of type attribute, degType. The dimension also contains BaseType{Degree} and subtype *research* or *coursework* which contains information of an area or a major in the degree. Finally, there is information regarding PK {DegreeID}. The integrated *Degree* dimension is defined as follows:

Dimension (Degree{Integrated Attribute, Inheritance Type, Nil})

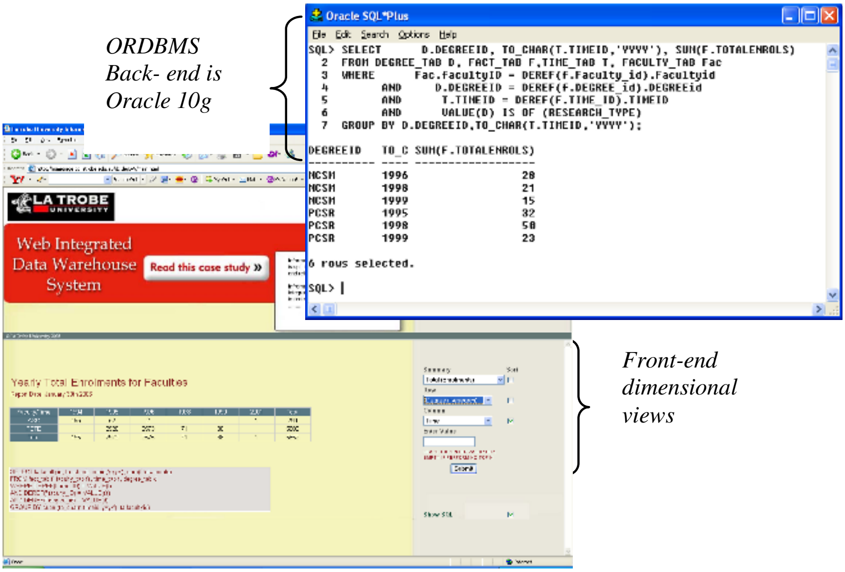
From Fig. 3, we specify important information for the Univeristy Fact class such as *Integrated attributes* {Timeid, Degreeid, Subjected, Facultyid, enrolments}; *Link type* {Association}; *PK/FK* {Degreeid, Subjectid, Facultyid, Timeid, enrolls}. The integrated Fact class is defined as follows:

Fact (uni_fact{Integrated Attributes, Association Type,One-to-many})

5.3 Specifying Complex Queries

The integrated web data warehouse model in Fig. 3. is translated into a physical integrated data warehouse system. A prototype of our case study has been developed in Oracle 10g. The underlying data sources for developing our prototype are described as the followings: (i) data of Health and Science faculty are stored in XML documents and (ii) data of other faculties are available in relational databases.

Fig. 6. shows a complex query to analyze the pattern enrolment in the Research degree type.



ORDBMS Back- end is Oracle 10g

```
SQL> SELECT      D.DEGREEID, TO_CHAR(T.TIMEID,'YYYY'), SUM(F.TOTALENROLS)
  2  FROM DEGREE_TAD D, FACT_TAD F,TIME_TAD T, FACULTY_TAD Fac
  3  WHERE       Fac.facultyID - DEREF(f.Faculty_id).Facultyid
  4        AND     D.DEGREEID = DEREF(f.DEGREE_id).DEGREEid
  5        AND      T.TIMEID = DEREF(F.TIME_ID).TIMEID
  6        AND      VALUE(D) IS OF (RESEARCH_TYPE)
  7  GROUP BY D.DEGREEID,TO_CHAR(T.TIMEID,'YYYY');

DEGREEID   TO_C SUM(F.TOTALENROLS)
---------- ---- ------------------
NCSM       1996                 28
NCSM       1998                 21
NCSM       1999                 15
PCSR       1995                 32
PCSR       1998                 58
PCSR       1999                 23

6 rows selected.

SQL> |
```

Oracle SQL*Plus
File Edit Search Options Help
LA TROBE UNIVERSITY
Web Integrated Data Warehouse System
Read this case study »
Yearly Total Enrolments for Faculties

Front-end dimensional views

Fig. 6. Complex Query Analysis

6 Conclusion, Discussion and Future Works

In comparison with existing integrated Web Data Warehouse techniques, our technique explicitly provides a fully integrated solution that covers from a conceptual design level to an implementation level. Our design phases are engaged with a very simple set of instructions to assist with the establishment of fact and dimensions in which hierarchies have been handled with wider choices of object oriented features.

This is not only to provide an impact understanding to all levels but also, at the implementation level, to efficiently store higher-level aggregated measures. Both Query processing and data retrieval could be considerably more efficient because of the leveling data representation.

In the completion of our proposed web data integrated approach, a complex query system that integrated data residing in a common web integrated data warehouse system is successfully accessed. In this context, the distinctive advantages in our proposed system that if there is no proper data integrated solution, information residing in multiple data source models need to be queried separately in order to compromise a specific requirement. In the process information retrieval is done on the individual system and aggregated measures are then recalculated using some alternates.

As for the future work, we would like to extend this paper to optimally ensure the quality and consistency by looking into data conflicts resolution during the integration process. Query processing performance may also be investigated.

References

[1] Bishay, L., Taniar, D., Jiang, Y., Rahayu, W., "Structured Web Pages Mamangement for Efficient Data Retrieval", WISE 2000, IEEE, pp. 97-104

[2] Buzydlowski, J. W., "A Framework for Object Oriented On-Line Analytic Processing, Data Warehousing and OLAP**",** ACM DOLAP 1998, ACM Press, pp.10 – 15

[3] Calvanese, D., Giacomo, G. De, Lenzerini, M., Rosati, D. N., "Source Integration in Data Warehouse", DEXA 1998, Springer, pp.192 - 197

[4] Chen, W., Hong, T., Lin, W., "Using the Compressed Data Model in Object-Oriented Data Warehousing", SMC 1999, IEEE, pp.768 – 772, Vol. 5

[5] Golfarelli, M., Rizzi, S., Birdoljak, B., "Data Warehousing from XML Sources", ACM DOLAP, ACM Press, pp.40 – 47

[6] Huang, S. M., Su, C. H.,"The development of an XML-based Data Warehouse System", ACM DOLAP 2001, ACM Press, pp.206-212

[7] Jensen, M., Moller, T., Pedersen, T., "Specifying OLAP cubes on XML data", SSDBM 2001, IEEE, pp.101 – 112

[8] Li, S., Liu, M., Wang, G., Peng, Z. "Capturing Semantic Hierarchies to Perform Meaningful Integration in HTML Tables", APWeb 2004, Springer, pp.899-902

[9] Mohamad, S., Rahayu, W., Dillon, T., "Object Relational Star Schemas", PDCS 2001, , IASTED

[10] Nassis, V., Rahayu, W., Rajugan, R., Dillon, T., "Conceptual Design of XML Document Warehouses", DaWak 2004, Springer, pp.: 1- 14

[11] Xyleme, L., "A Dynamic Warehouse for XML data of the Web" IDEAS 2001, IEEE-CS, 2001, pp.3–8

Location Aware Business Process Deployment

Saqib Ali, Torab Torabi, and Hassan Ali

Department of Computer Science and Computer Engineering,
La Trobe University,
Bundoora, VIC, 3086, Australia
{saqib.ali, t.torabi}@latrobe.edu.au,
h5ali@students.latrobe.edu.au

Abstract. Every action a business process performs must be explicitly anticipated, designed for and implemented by business professionals. Most of the current techniques specify business processes (BP) without incorporating all four Ws; **W**ho, **W**hen, **W**hat and **W**here. These processes when used especially in logistics or supply chain applications will result in a BP becoming even more complicated and harder to customize. The business process is dependent upon business rules (BR), its resources to achieve its objectives. To overcome some of these issues we propose a location aware business process deployment framework. Using this framework we can integrate location awareness into the existing business processes. In this paper our focus would be on how the companies can adopt for their traditional business processes to be mobile. We have developed a case study using location aware methodologies into existing process for development of a more efficient and effective enterprise application.

1 Introduction

Business process is defined as a set of linked procedures or activities, which collectively realize a business objective or policy goal. This is normally within the context of an organizational structure defining functional roles and relationships [1]. The software process usually specifies the actors executing the activities, their roles and the artifacts produced [2]. Businesses realize that the cost of automating transactions with trading partners is very high. Standards and technologies for modeling business process that use web services could drive the costs down by achieving automated business process [2], [3]. Traditional applications cannot support the flexibility of location dependency in business process.

As the technology is changing very fast and the companies operate in complex environments that consist of thousand of processes, the business profit depends on efficient delivery of goods and services controlled by business process [4], [5]. So there is a need for the companies to make use of the technologies to make their product more profitable and their services more efficient.

Now many companies are earning profits by using mobile technology in their business applications. By this these companies tend to make their traditional business processes into mobile business processes. "*Mobile Business Process*" is a business process, when the place of execution of an activity can be different in different instances of the business process or the places can change during the execution of an activity [6], [7].

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 217 – 225, 2006.

The use of the Mobile Technology in business application has helped the companies to reduce costs and provide new revenues by improving business processes, creating competitive advantages, improving the efficiency of the mobile workforce, and by guiding stakeholders to maximize their efficiency thus reducing field costs [8], [6].

There is a need to make business process more location aware. Business processes also needs to be customizable and reusable so that the companies can be able to work more efficiently and be able to provide better service to the customers. In this paper we propose a location aware business process framework which would be used to make a mobile business process more location aware. This would result in a more complete, accurate, and flexible use of the Business Process.

2 Proposed Location Aware Business Process Framework

A location-aware application makes use of a user's location. A Location aware application is a middleware that lets company's business application take advantage of location based services from multiple vendors, while providing application developers with an easy-to-use, yet powerful Application Programming Interface (API) [7]. In this section we propose a location aware business process framework. We divide the proposed framework into different environments. We simply grouped similar functionalities into an environment, for example mobile and non mobile services in separate environments. Each environment has its own associated behavior and characteristics. Figure 1 shows that mobile and non-mobile environments perform their business processes through server environment, which is also the communication layer between business process environment.

We use software agents [9], [10], [11] to synchronize, integrate and execute all the business processes which are defined in business process environment for both mobile

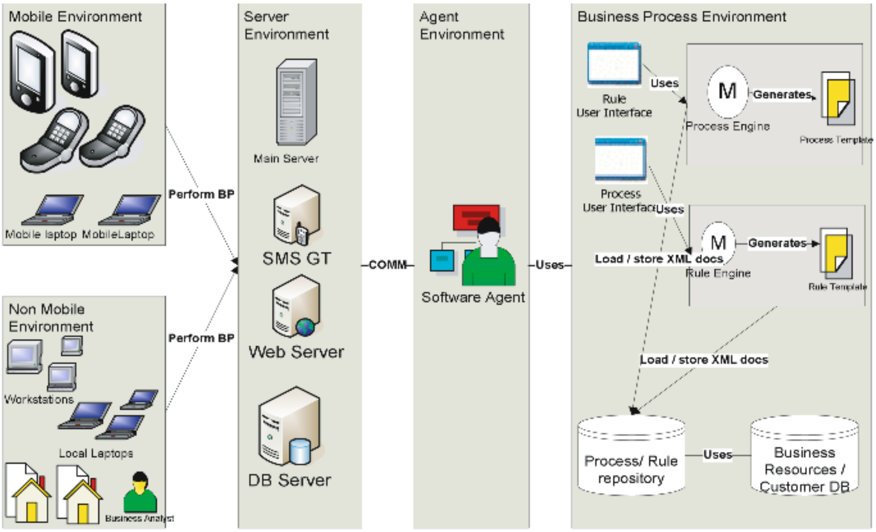


Fig. 1. General overview of mobile business process framework

and non mobile devices. We have identified business process environment as a core component of this framework and have separated from the other components. Our framework uses agent oriented rule-based approach [12]. We separate agent environment from main business process environment, the agent's behavior and their actions controlled and customized through this environment. In business process environment business logics and process are defined. Using the framework it is easy for a business to introduce and integrate new process in this environment.

In following sub-sections we present an overview about different environments used in our framework.

2.1 Mobile and Non-mobile Environment

Mobile and non-mobile devices helps to determine the location of a process. All mobile and non-mobile devices are assessed through server environment. All static entities are come under non-mobile environment and an entity whose location is not static is come under mobile environment. Both environments accessed through server environment. Non-mobile environment is used to provide a flexibility of executing business processes through web pages. For example a customer can put a pickup request through company's web pages. Whereas mobile environment is use to provide a link between business processes and non-mobile devices. Business analysts also use these environments to define its business process or process conditions. Latest technologies like GPRS (General Packet Radio Service) are used to detect the location of mobile devices.

2.2 Server Environment:

Figure 2 shows different sub servers that are connected with main server for connectivity with external entities like mobile or non mobile devices. The sub servers may

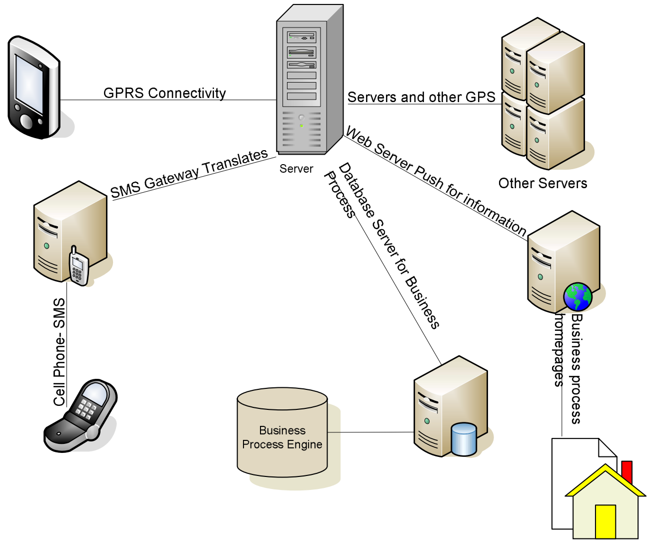


Fig. 2. Communication layout of business process

include web servers, SMS gateway translators, database servers etc. The main server is also used for GPRS (General Packet Radio Service) connectivity.

This environment is used to control all the communication connections used in location aware approach. There are different ways or methods are used to access any business process depended upon different locations including use of web services. Every server entity is linked with main server, which is responsible for accessing authority.

2.3 Agent Environment

Agent environment is a middle-ware between mobile and non mobile devices. Agent is responsible for synchronizing business processes, their integration with rules and deployment or execution of business processes within its environment shell depending upon business resources. The agent performs all its actions with its own defined ADDED properties [12], [13]. We are using software agents to handle all the business process's synchronization, integration and executions.

The agent is situated in a business environment with set goals, abilities to perform actions and having understanding of environmental characteristics [10]. An agent is capable for automating new processes at different locations depending upon roles, rules and companies resources. This agent has the objective (set goals) of calculating (performing actions) all the new processes with new process conditions (business rules) within business process environment.

2.4 Business Process Environment

This is a main environment where business processes are defined. In this environment business process and rule engines are defined where user can define business processes as well as process's rules and conditions. For simplicity rule and process

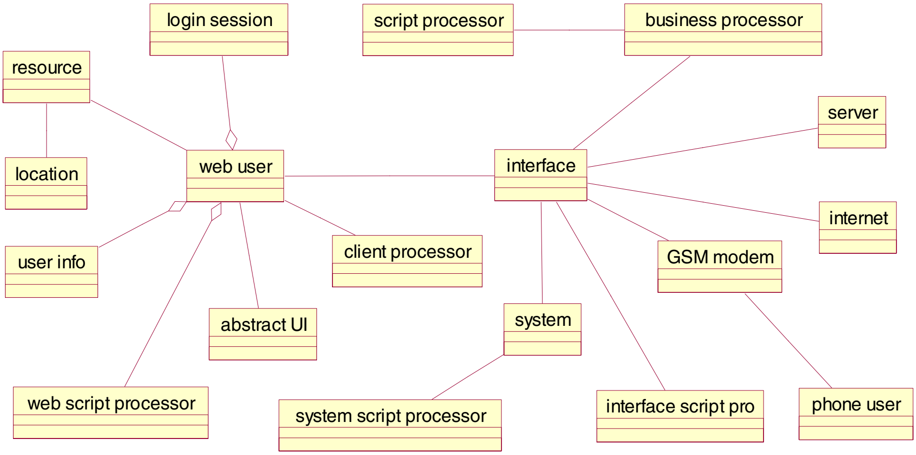


Fig. 3. Classes of location aware business process

engines are not discussed in this paper, we adopt agent oriented rule based business process approach [12], [14], [15]. Business resources and processes repository is used to define or customize business processes. Business analyst can use mobile or non mobile devices to customize existing processes through process engines or alternatively customize conditions for any particular process.

Figure 3 (class Diagram) shows the different classes and their relationships used in the location aware business process approach. This class diagram presents the overall representation of the application in the context of the location aware approach. Classes are defined in such a way that user can interact with system either from mobile technology or through web based applications.

Following are brief overview of some of the classes used in our approach:

Web user is the class for the user who will use the system through the website. This will be in regard to our system where the user will do its activities through the website.

Loginsession class is defined to record all the session details for the specific webuser. Userinfo class is used to record user information (such as address, phone etc.). Abstract UI class is being used to store the user interface customization layer, where each user will be able to customize his/her view. Every user would have specific profiles within the company.

Script processors

Script processors include WebuserScriptProcessor, SystemScriptProcessor, and InterfaceScriptProcessor classes. These classes are used to store all the functionalities and commands as scripts in the database. These script processor classes process the scripts from the database that are going to be used in the system.

Interface class

Interface class connects the mobile user to the rest of the system. A mobile user may not be able to connect to the system directly, as there are issues in mobile systems such as the device compatibility, and service integration, etc. All these issues are taken into consideration by interface class. Script class is also connected to the interface class that processes the scripts from the database class.

Business processor class

Business processor class stores different process or activities which are stored as scripts in the database. To run these scripts from the database we need the script processor that is also connected to the database.

Resource and location class

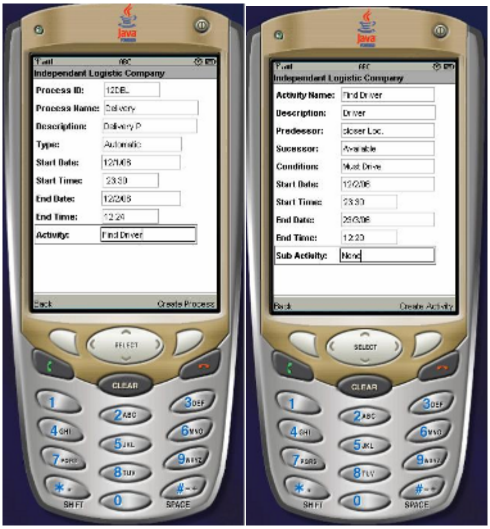
The resource class is used to have all the information associated with company's resources. And location class is part of resource class, which is responsible for having all the information about which resources are needed at what location by whom and when.

3 Case Study

This case study presents a scenario for a company that uses the location aware technologies and information intensive business processes to enrich its existing enterprise applications such as service dispatch or fleet management. The company has different processes like order pickups, order delivery and invoicing. All these processes are being processed at different locations. Each location may have different set of processes and activities. Each particular process (**What**) is done by particular actor (**Who**), at particular time (**When**) and at particular location (**Where**).

The company has fleets that are being used to pickup order from one location and deliver to another. The movement of the fleet is limited to the confined limits of the city or a state. The Application starts when a customer uses his/her mobile or web based interface to login and request for a package to be dispatched by entering its pickup and destination addresses and other related information. Then the dispatcher server would track the position of its fleet as well as their status (**idle**, **busy** or **off-duty**) and would determine the travel time between customer's pickup location and the current location of the fleet show with status as idle. If the distance of the idle fleet and the customer require a long delay before pickup, then the dispatch server would automatically allocate the pickup to a busy fleet. Based on conditions and the parameters the best suitable fleet is chosen in regard to the proximity of the location. A company can use its business processes in a more efficient manner. This could also lead to the greater increase in the company's revenue and a reduction in the losses.

We have implemented Independent Logistic System (ILS) using latest JAVA technologies. We adopt one scenario of independent logistics business process case study where customer request for delivery of plastic sheets (product) at particular location.



Independendant Logistic Company

Process ID: 12DEL
Process Name: Delivery
Description: Delivery P
Type: Automatic
Start Date: 12/1/08
Start Time: 23:30
End Date: 12/2/08
End Time: 12:24
Activity: Find Driver

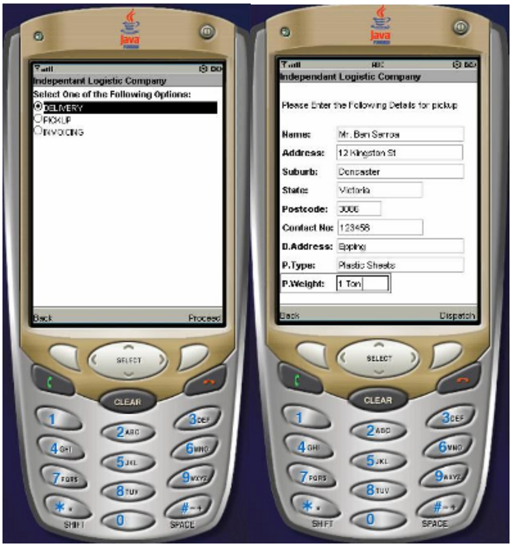
Independendant Logistic Company

Activity Name: Find Driver
Description: Driver
Predessor: closer Loc.
Sucessor: Available
Condition: Must Drive
Start Date: 12/2/06
Start Time: 23:30
End Date: 29/3/06
End Time: 12:20
Sub Activity: None

Fig. 4. (a) defining delivery process (b) defining an activity

First we define a delivery process for this particular system. In figure 4(a), business analyst define a process through an emulator by entering relevant information like process id, name, description, its type, start date and so on. Every process may or may not have sub activity. In delivery process sub activity is to find suitable driver for pickup and delivery whose status is available. Figure 4 (b), shows the activity of delivery process.

In figure 5, customer interacts with different options such as making a request for pickup, delivery or invoicing. After selection, customer enters all relevant details including pickup, drop-off location, product type, weight and contact details in the system. The system will automatically dispatch his/her request through his/her mobile device.



Independent Logistic Company

Select One of the Following Options:

● DELIVERY
○ PICKUP
○ INVOICING

Back Proceed

Independant Logistic Company

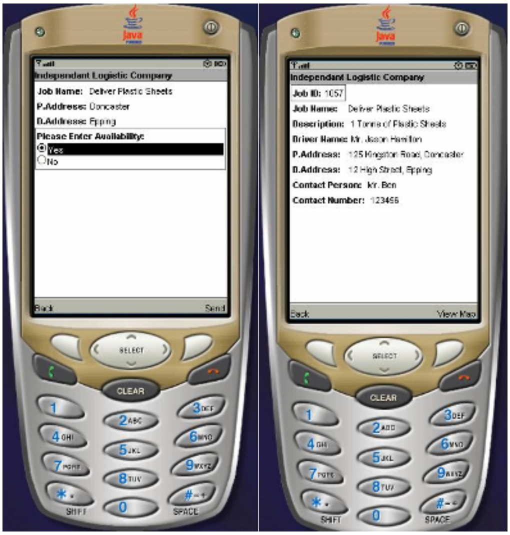
Please Enter the Following Details for pickup

Name:	Mr. Ben Serros
Address:	12 Kingston St
Suburb:	Doncaster
State:	Victoria
Postcode:	3006
Contact No:	123456
D.Address:	Epping
P.Type:	Plastic Sheets
P.Weight:	1 Ton

Back Dispatch

Fig. 5. Initial customer request

Once the request is placed, the system will locate a suitable driver by fulfilling process conditions. The system uses global positioning system to locate all the available drivers within that particular range whose status are idle or available. The driver's information is stored within the system database. Once the system locates a driver then it sends a message to driver for his acceptance of a particular job. It might be possible that the available driver is not willing to accept this job; system will provide basic information about delivery job on his mobile device. Driver will only see pickup and drop off location (suburb only) and he is given two options either to accept or decline a job. Once the driver accepts this job, the system will provide all the detail information for him to perform a task. Figure 6 (a) and (b) shows the emulator's screen displayed to the driver before and after accepting the job.



Independant Logistic Company

Job Name: Deliver Plastic Sheets

P.Address: Doncaster

D.Address: Epping

Please Enter Availability:

● Yes

○ No

Back Send

Independant Logistic Company

Job ID: 1057

Job Name: Deliver Plastic Sheets

Description: 1 Tonne of Plastic Sheets

Driver Name: Mr. Jason Hamilton

P.Address: 125 Kingston Road, Doncaster

D.Address: 12 High Street, Epping

Contact Person: Mr. Ben

Contact Number: 123456

Back View Map

Fig. 6. (a) Job initial dispatch (b) Job details

4 Conclusion and Future Work

In this paper we have discussed a framework that is used to implement and integrate the business processes into location aware environment. In our approach, we have divided the framework into different environments. Each environment has its own associated behavior and characteristics. All the similar functionalities are grouped into environments like mobile and non-mobile devices in one environment. Mobile and non-Mobile environments perform their business processes through server environment, which provides a communication layer between business process environments. We have introduced software agents to synchronize, integrate and execute all business processes that are defined in business process environment used for both mobile and non-mobile devices. We have proposed and implemented a case study on independent logistic company (ILS), in which we presented a practical application to support our theoretical research findings. By using our framework, businesses are able to handle business processes more efficiently into complex business process automation where business is dependent upon location information.

For Future work, we will be extending our location aware business processes into context aware. In the current mobile technology, context-aware applications are only restricted to location aware concept for mobile applications (Location Based Services). Our future work also involves deployment of a business process to a mobile application according to the indicators of the context-awareness such as identity, schedules, agenda settings, activity (talking, walking and running), and availability of resources.

References

1. Dyal, U., M. Hsu, and R. Ladin. "Business Process Coordination: State of the Art, Trends, and Open Issues." in *Proceedings of the 27th. VLDB conference, Roma Italy*: 2001.
2. D.Rosca, and C.Wild, "Towards a Flexible Deployment of Business Rules". *Expert Systems with Applications*, 2002, p. 385-394.
3. Garicano, L. and S.N. Kaplan. "The Effects of Business to Business eCommerce on Transaction Costs: Description, Examples and Implications". in *Working Paper*: 2000.
4. Sloser, S.E., "Optio e.ComEngine and Optio e.ComIntegrate - From Optio Software". *First Impression*, 2000, p. 88; S.Ali;, B.Soh;, and J.Lai. "Rule extraction methodology by using XML for business rules in a business process". in *IEEE 3rd. International conference on industrial informatics (INDIN 2005)*. Perth, Australia: 2005. p. 107-111.
5. Willis, D., "Twenty Questions on Virtual Trading communities for IPNet Solutions' CEO". *EAI Journel*, 2000, p. 108 - 110; S.Ali;, B.Soh;, and T.Torabi. "The new adaptive business model for e-commerce". in *Third International conference on Information Technology (ICITA 2005)*; IEEE;. Sydney, Australia: 2005. p. 99-102.
6. Tilson, D., K. Lyytinen;, and R. Baxter. "A Framework for selecting a location based service (LBS) strategy and service portfolio." in *Proceedings of the 37th Hawaii International conference on system sciences.*; IEEE. Hawaii, USA.: 2004. p. 1-10.
7. Houssos, N., V. Gazis;, and A. Alonistioti, "Enabling Delivery of Mobile Service Over Heterogeneous Converged Infrastructures". *Information systems Frontiers*. 6(3), 2004, p. 189-204.
8. Barkhuus, L. and A. Dey. "Is Context-aware computing taking control away from the user? Three levels of Ineractivity Examined." in *Proceedings of the Fifth annual conference on Ubiquitous Computing (UBICOMP 2003)*: 2003. p. 1-9.
9. N.R.Jennings, "In agent-based software engineering". *Artificial Intelligence*, 2000, p. 277-296.
10. G.Wagner. "Agent-oriented Enterprise and business process modelling." in *First international workshop on Enterprise Management and Resource Planning Systems*. Venice: 1999.
11. Wagner, G., Y. Lesperance;, and E.Yu. "Agent-orientd information systems". in *2nd. International workshop at CAiSE*00*; iCue. Stockholm, Berlin: 2000.
12. S.Ali, B.Soh, and T.Torabi. "An agent oriented framework for automating rules in business process". in *IEEE 3rd. International conference on industrial informatics (INDIN 2005)*. Perth, Australia: 2005. p. 17-21.
13. G.Weiss, "Multiagent systems: a modern approach to distributed artificial intelligence". Cambridge, Mass; MIT press, London: 1999.
14. T.Torabi, S.Ali, and W.Rahayu. "A specification for Business Model Components for B2B communication". in *IASTED Software Engineering Application Conference*. Marina del Ray, USA: 2003. p. 459-464.
15. S.Ali, B.Soh, and T.Torabi. "Using software engineering principles to develop reusable business rules". in *1st. international conference on information and communication technology (ICICT 2005)*. Karachi, Pakistan: 2005. p. 276-283.

A Framework for Rapid Development of RFID Applications[*]

Youngbong Kim, Mikyeong Moon, and Keunhyuk Yeom

Department of Computer Engineering, Pusan National University,
Pusan, 609-735, Republic of Korea
{saram, mkmoon, yeom}@pusan.ac.kr

Abstract. Radio Frequency IDentification (RFID) technology is considered to be the next step in the revolution in supply-chain management, retail, and beyond. To derive real benefit from RFID, a RFID application must implement functions to process the enormous event data generated quickly by RFID operations. For this reason, many RFID middleware systems have been developed. Although RFID middleware assists in the management of the flow of event data, developers will be forced to implement systems to derive from simple RFID events meaningful high-level events, which are more actionable knowledge that can be applied. Determining meaningful events requires a context, which typically comes from reference data. In this paper, we propose the contextual event framework (CEF) for rapid development of RFID applications. The solutions and techniques presented in this paper are based on our experience of RFID middleware of the Logistics Information Technology (LIT) project.

Keywords: RFID, contextual event, RFID application development framework, RFID event.

1 Introduction

Radio Frequency IDentification (RFID) is the general term for technologies that identify a person or object using radio frequency transmission [1]. At the point where the RFID reader receives the signal emanating from a tag and then passes the data to applications and services, RFID technology is similar to bar code scanning. However, unlike the more traditional bar code scanning, RFID does not require "line-of-sight" for readers to receive the information contained on small tags and a reader can detect multiple tags simultaneously. With these significant advantages, RFID technology is considered to be the next step in the revolution in supply-chain management [2, 3, 4, 5], retail [6], and beyond [7, 8].

The character of the RFID event and the flood of information it generates make efficiency in dealing with it very important. RFID events are generated quickly and

[*] We would like to express heartfelt thanks for God's support to this paper. This work was supported by the Regional Research Centers Program(Research Center for Logistics Information Technology), granted by the Korean Ministry of Education & Human Resources Development.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 226 – 235, 2006.

the volume of RFID events can be enormous. In addition, most RFID data are simple. Unless we are using sophisticated, expensive tags, all we receive is an identification number for the item, a time, and a location [9]. In early RFID solutions, sensor readings were sent directly to the applications and services; therefore, the applications and services had to interpret the preliminary readings. This approach is very complex for interpreting RFID events and is neither scalable nor adaptable [10]. Recently, many RFID middleware systems have been developed by major corporations [11, 12, 13, 14]. Although RFID middleware deletes duplicate readings from the same tag and helps manage the flow of data, developers will be forced to implement systems to derive meaningful high-level events, which contain more actionable knowledge for application than the simple RFID events.

In this paper, we propose the contextual event framework (CEF) for rapid development of RFID applications, including its core concepts of contextual event, contextual event language, and contextual event assistant. A contextual event is a derivation of simple RFID events. It is the meaningful high-level event that contains more actionable knowledge for application. We describe CEL, an XML-based Contextual Event Language, which shields the details of the underlying RFID infrastructure and allows programmers to specify contextual event in an intuitive way. It has been carefully designed to include a set of essential activities to simplify the specification of RFID applications. The contextual event assistant (CEA) processes a chain of the activities. We have developed the CEF and applied it to an application development. The power of this approach is illustrated by a case study.

This paper is organized as follows: Section 2 describes the basic concepts of a contextual event. Section 3 presents the CEL and section 4 presents the CEA, which are the two core sections of the CEF. Section 5 describes the current implementation status of CEF, how CEF has been used, and describes a project. Related studies, conclusions and suggestions for future work are given in sections 6 and 7.

2 Basic Concepts of the Contextual Event

This section explains concepts that form the foundation of the contextual event framework, the *RFID event* and *contextual event*.

An *event* is defined as an object that is a record of an activity in a system [15]. An event may have particular data components. Data components of an event can include the time period of the activity, where it happened, who did it, and other data. We classify the events that occur in an RFID system according to the definition of an event.

2.1 RFID Events

Fig. 1 shows the architecture, layers, and the events (reader event and RFID event) in an existing RFID system. The RFID system is generally partitioned into three tiers: the reader layer, the RFID middleware layer, and the application layer. RFID middleware receives reader event streams (tags) from one or more RFID readers. It collects, filters, and cleanses these reader events to make them available to the RFID applications. As shown in Table 1, an RFID event is defined as an event caused by

RFID middleware. Included in its information are the logical reader name, tag value, direction, and time. The application developer must collect RFID events, access the data server to retrieve reference data of RFID events, and process business logic (rules) to implement the RFID applications. That is, application developers must be conversant with RFID knowledge and communication techniques; substantial applications should involve additional codes, rather than just business logic, to process RFID events.

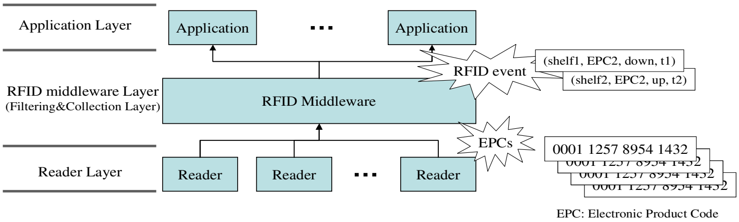


Fig. 1. RFID event in architecture of RFID system

Table 1. RFID event definition

event	producer	consumer	definition
RFID event	RFID middleware	Applications	Filtered reader event <logical reader name, EPC, up/down, time>

2.2 Contextual Events

Fig. 2 shows the role of the RFID CEA in an RFID system and the types of events that can occur within the system. The CEA receives RFID events from the RFID middleware, converting them to a more actionable form that can be effectively used by the RFID application.

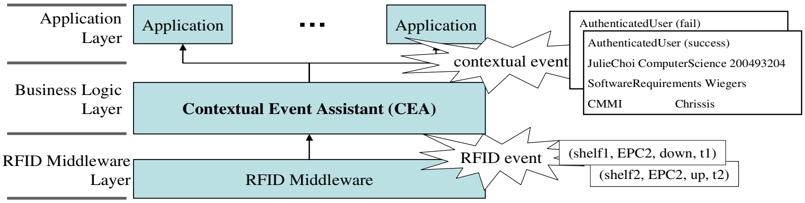


Fig. 2. Role of contextual event assistant

A contextual event is a higher-level event indicating use of various activities in the application. A contextual event is defined as that which is derived from the simple RFID event. Conceptually it is combined with an RFID event, reference data, and business rule. The difference between an RFID event and a contextual event is obvious from responses to questions such as the following.

Questions:

- For an RFID event: “What is at Reader A now?”
- For a contextual event: “Is the person authenticated who is captured at Reader A now?”

Responses:

- For an RFID event:
  <dock_A urn:epc:tag:sgtin-96:4.011562.0557083.19212150 up 10:12:00:06:05>
- For a contextual event:
  <authenticatedUser(success): ID 9034 Name Kimwoo type Manager>

Elements of the second question, for a contextual event, are interpreted as follows:

- RFID event: “The tag value captured at Reader A”
- Reference data: “The identity of the person with the tag”
- Business rule: “The corresponding authentication rule”

Additionally, this event has data components. Table 2 indicates the definition and the form of the contextual event.

Table 2. Definition of contextual event

event	producer	consumer	definition
Contextual event	CEA	Applications	Derived RFID event <contextual event name(the result of the corresponding business rule):[data component]*>

The CEA is a means for achieving transformation from the RFID events to the contextual events. The transformation processes consist of a small number of activities that collect RFID events, retrieve reference data, analyze the corresponding business rule, and generate contextual events. To produce a contextual event, a specification (CESpec: Contextual Event Specification) should be describded for these activities. The CEL for describing the CESpec is described in the next section.

3 Contextual Event Language (CEL)

CEL is an XML-based language for describing the contextual events at a high level of abstraction without dealing with implementation detail. In CEL, an *activity* is a generic unit of work, specified either as a *declaration activity* for defining the data variable, a *trigger activity* for collecting RFID events, a *reference activity* for retrieving reference data or a *generation activity* for generating contextual events and related data.

3.1 Declaration Activity

Variables in a CESpec must be declared along with their type. The type may be a general data type, such as *integer, float,* and *string*, or may be an RFID specific data type, such as *EPCtag* and *EPCtagList*. *EPCtag* is a data type that stores a product code value (tag value). *EPCtagList* is a data type deals a list of tags produced from RFID middleware in one event cycle.

3.2 Trigger Activity

CESpec is triggered by one or more RFID events. *Trigger activity* defines processes that request RFID events. It consists of elements that can describe the RFID events of interest and the RFID reader control information related to the events, such as a start or stop trigger for an event cycle, repeat period and duration. In addition, it defines an element that receives RFID events from the RFID middleware.

3.3 Reference Activity

To transform an RFID event into a contextual event, CESpec should be described in terms of the required context, which typically comes from the reference data. The reference data are retrieved from an information service. An EPC Information Service (EPCIS) is the networked database that stores the additional data associated with the tagged object. It provides a standard interface for access and persistent storage of EPC-related data [16]. *Reference activity* defines processes to retrieve data from EPCIS.

3.4 Generation Activity

Generation activity is composed of a *condition* and a *generation*.

- Condition

  The condition verifies the trigger that CESpec requires to react to the triggering RFID events. It represents a business rule, which is required in the applications. The business rule constrains some aspect of the business related to the RFID event and the reference data.
- Generation

  The generation defines processes that notify the subscribed contextual events to the application. The contents of the notification include contextual event name, the result of the corresponding business rule, and a related data component.

4 Contextual Event Assistant (CEA)

The CEA provides the means of processing contextual events that are described as CEL. Each activity in the CEL is mapped to components in the CEA. It is designed for use in a middleware-based RFID system. Next, we describe the role of the CEA in an RFID system and the architecture of the CEA.

4.1 CEA in an RFID System

The role of the CEA in an RFID system is shown Fig. 3. The RFID middleware receives raw RFID events from the RFID readers and converts them into a form that can be used by the CEA. The CEA transforms RFID events to contextual events using a reference data, which is retrieved from the EPCIS, object-naming service (ONS) or other Reference Data Server. EPCglobal, the current EPC standard group, defines the ONS and EPCIS to exchange product-level information [16, 17] in the networks for RFID data and product data [18].

An RFID application sends requests to the CEA for contextual events. The application uses the client API of the CEA to send CESpecs to the CEA and to receive Contextual Event Reports (CEReports) from the CEA.

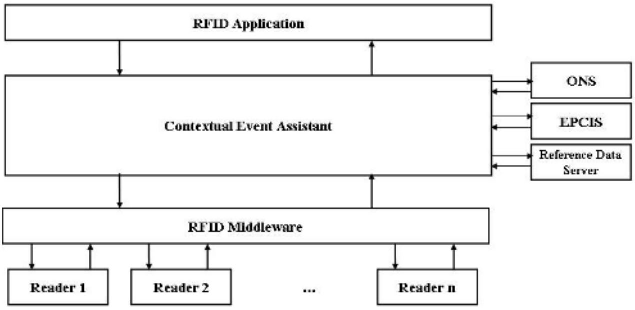


Fig. 3. CEA in an RFID system

4.2 CEA Architecture

As shown in Fig. 4, the CEA consists of the following components:

- *Contextual Event Notification.* This component receives a contextual event request sent by the RFID application and, once the requested contextual event is detected, sends the events to the subscribing applications.
- *Contextual Event Management.* Collected RFID events are transformed to contextual events using reference data and a corresponding business rule in this component. Activity composition is the responsibility of the component. The component decomposes the CESpec passed for registration and finds adequate activities.
- *RFID Event Collection.* The CEA does not receive events directly from RFID readers. Using the interface, which RFID middleware provides, this component specifies which RFID events are of interest and RFID reader control information related to the events, such as a start or stop trigger for an event cycle, repeat period and duration. In addition, it receives RFID events from RFID middleware.
- *Reference Data Collection.* This component receives reference data, which are required to produce contextual events from external data servers. This component consists of the EPCIS access and the ONS access components, enabling it to collect data from the EPCIS and the ONS. If we wish to collect

reference data from other data sources, we can develop our own collection adapter-reference data server access component for reference data and can use this after registration.

- *CESpec Registry/CE Repository.* Registered contextual event specifications are stored in the *CESpec Registry* and contextual events generated by the *Contextual Event Management* component are stored in the *CE Repository*.

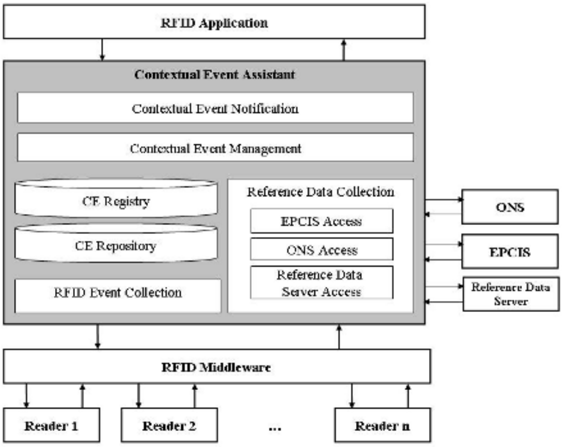


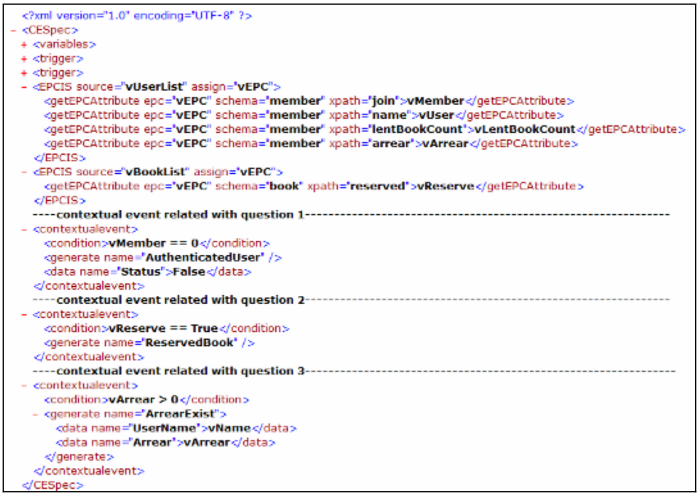
Fig. 4. CEA Architecture

5 Experiments

Logistics Information Technology (LIT) is the Korean national project for developing the next generation of logistics information technology. The research center for LIT has developed the prototype, version 1.0, of the LIT RFID system. In particular, the prototype of the CEF was developed and demonstrated with other components (the RFID middleware, the EPCIS and the ONS) developed at the LIT project. Our proposed approach is applied to the development of PNU library system, blood management system, and store management system.

In this section, we describe the use of CEF in the development of PNU library system. The PNU library system automatically processes the loan and return services using the LIT RFID system. At each point along the library services chain, an RFID reader "reads" the tags that are fixed to the books, librarians, and authenticated users. The readers send the tag values to the RFID middleware, making them available to the CEA. The CEA transformed the RFID events into contextual events in accordance with the CESpec and returns generated contextual events to the applications. So it was not necessary for the applications to strive to process the RFID events. Examples of question for contextual event and a CESpec (Fig. 5) in the loan service are:

Question 1: *Is the person authenticated who is captured at Reader A now?*
Question 2: *Is it possible to lend the book that is captured at Reader A now?*
Question 3: *Is it possible for the authenticated person to borrow the books that are captured at Reader A now?*



```
  <?xml version="1.0" encoding="UTF-8" ?>
- <CESpec>
  + <variables>
  + <trigger>
  + <trigger>
  - <EPCIS source="vUserList" assign="vEPC">
      <getEPCAttribute epc="vEPC" schema="member" xpath="join">vMember</getEPCAttribute>
      <getEPCAttribute epc="vEPC" schema="member" xpath="name">vUser</getEPCAttribute>
      <getEPCAttribute epc="vEPC" schema="member" xpath="lentBookCount">vLentBookCount</getEPCAttribute>
      <getEPCAttribute epc="vEPC" schema="member" xpath="arrear">vArrear</getEPCAttribute>
    </EPCIS>
  - <EPCIS source="vBookList" assign="vEPC">
      <getEPCAttribute epc="vEPC" schema="book" xpath="reserved">vReserve</getEPCAttribute>
    </EPCIS>
    ----contextual event related with question 1--------------------------------------------------------------
  - <contextualevent>
      <condition>vMember == 0</condition>
      <generate name="AuthenticatedUser" />
      <data name="Status">False</data>
    </contextualevent>
    ----contextual event related with question 2--------------------------------------------------------------
  - <contextualevent>
      <condition>vReserve == True</condition>
      <generate name="ReservedBook" />
    </contextualevent>
    ----contextual event related with question 3---------------------------------------------------------------
  - <contextualevent>
      <condition>vArrear > 0</condition>
    - <generate name="ArrearExist">
        <data name="UserName">vName</data>
        <data name="Arrear">vArrear</data>
      </generate>
    </contextualevent>
  </CESpec>
```

Fig. 5. Example of CESpec

We demonstrated that the contextual events can be sufficiently specified using CEL, and our CEA can effectively provide the contextual event. Additionally, we confirmed that CEA reduces the development effort for the RFID application by comparing the code sizes of the CEA-based RFID application and the RFID application without CEA.

6 Related Studies

EPCglobal [19], a not-for-profit standards organization that is commercializing and driving the global adoption of Electronic Product Code (EPC) technology, defines the network associated with RFID data and product data [16, 17, 18, 20]. It developed the standard for transforming a raw tag-reading stream into RFID events. However, contextual event processing is not a task it is considering.

Vendors like Sun Microsystems [11], IBM [12], Oracle [13], and Microsoft [14] have been extending their application development and middleware technology stacks to handle RFID requirements. These middleware systems delete duplicate readings of the same tag and help manage the flow of data. Most of RFID events generated by these RFID middlewares are too simple for applications. Unless we're using sophisticated, expensive tags, all we get is a location, an identification number for the item, direction and a time [9] (e.g., dock_A urn:epc:tag:sgtin-96:4.011562.0557083.19212150 up 10:12:00:06:05). Developers will be forced to implement systems to derive meaningful high-level event (e.g. authenticatedUser (success): id 9034 name Kimwoo type Manager), from simple RFID events.

There are several researches about deriving meaningful context information from raw data acquired by sensors. Recent research work has focused on providing infrastructure support for context-aware system. Ranganathan and Campbell proposed a middleware that facilitates the development of context-aware agents [21]. Reconfigurable Context-Sensitive Middleware facilitates the development and runtime operations of context-sensitive pervasive computing software [22]. Tao gu developed a service-oriented middleware provides support for acquiring, discovering, interpreting and accesses various contexts to build context-aware services [23]. These middleware are for general sensors, thus do not address various characteristics of RFID technology.

7 Conclusions and Future Work

The RFID middleware extracts data from the RFID reader, filters them, combines the information, and routes the RFID events to the applications. Although RFID middleware helps manage the flow of event data, developers will be forced to implement applications to derive meaningful high-level events, which are more usable in applications, from the simple RFID events. In this paper, we proposed the CEF, including its core concepts of contextual event, contextual event language, and contextual event assistant. A contextual event is a meaningful high-level event that is more usable in the application than the simple RFID events. To specify contextual events, the CEL is described. The CEL is composed of a number of activities that collect RFID events, retrieve reference data, analyze the corresponding business rule, and generate contextual events. The CEA processes a chain of these activities that control processing the contextual event request.

By using CEF, applications do not have to involve additional codes to process RFID events, thereby substantially reducing the cost of developing and managing RFID applications. In addition, CEF enables developers to view RFID applications in terms of how they use contextual events — not in terms of how they build contextual events.

Our future research activities include the development of tools to support and analyze contextual events and describe CESpecs. In addition, we will research a workflow that can manage business processes triggered by the contextual events.

References

[1] A Basic Introduction to RFID technology and Its use in the supply chain, http://www.printronix.com/uploadedFiles/Laran_WhitePaper_RFID.pdf, January 2004.

[2] Walmart Supplier Information: Radio Frequency Identification Usage. http://www.walmart stores.com, 2005.

[3] DoD RFID Official Website. http://www.dodrfid.org.

[4] Teresko, J., "Winning with Wireless", Industry Week, 252(6), http://www.industryweek.com/CurrentArticles/Asp/articles.asp?ArticleId=1434, June 2003.

[5] H.K. Launches RFID Supply Chain Project, http://www.rfidjournal.com/article/articleview/ 1630/1/1/, June 2005.

[6] Retailer RFID Spending Projected To Research $4.2 Billion, http://www.techweb.com/wire /172303296, October 2005.

[7] Group Finalizes Drug Security Network, http://www.rfidjournal.com/article/articleview/15 85 /1/8/, May 2005.

[8] Chang-Gung Memorial Hospital, http://www.cgmh.org.tw/eng2002/intr_kel.htm

[9] M. Palmer, Seven Principles of Effective RFID Data Management, http://www.objectstore. com/docs/articles/7principles_rifd_mgmnt.pdf, August 2004.

[10] Fusheng Wang and Peiya Lie, "Temporal RFID Data Management", In VLDB, pages 1128-1139, 2005.

[11] Sun Microsystems, http://www.sun.com/software/solutions/rfid/

[12] IBM, http://www306.ibm.com/software/pervasive/w_rfid_premises_server/, December 2004.

[13] Oracle, http://www.oracle.com/technology/products/iaswe/edge_server

[14] Microsoft, http://www.microsoft.com/business/insights/about/aboutus.aspx

[15] Luckham, D., *The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems*, Addison-Wesley, ISBN 0-201-72789-7, 2002.

[16] Mark Harrison, "EPC Information Service", January 2004.

[17] EPCglobal, EPCglobal Object Name Service (ONS) 1.0 Working Draft Version, April 2004.

[18] EPCglobal, The EPCglobal Network and The Global Data Synchronization Network (GDSN): Understanding the Information & the Information Networks, October 2004.

[19] EPCglobal, http://epcglobalus.gs1us.org/

[20] EPCglobal, The Application Level Events (ALE) Specification Version 1.0, September 2005.

[21] Ranganathan, A. and Campbell, R.H, "A Middleware for Context-Aware Agents in Ubiquitous Computing Environments", Lecture Notes in Computer Science, Vol.2672, Springer-Verlag, 2003.

[22] Stephen S. Yau, Fariaz Karim, Yu Wang, Bin Wang, and Sandeep K.S. Gupta, "Reconfigurable Context-Sensitive Middleware for Pervasive Computing", In IEEE Pervasive Computing, July-September 2002.

[23] Tao Gu, Hung K.P, and Da Q.Z, "A Service-oriented middleware for building context-aware services", Journal of Network and Computer Applications, Vol.28, 2005.

A Flexible DRM System Considering Ubiquitous Environment*

Jong Hyuk Park[1], Sangjin Lee[1], and Byoung-Soo Koh[2]

[1] Center for Information Security Technologies, Korea University,
5-Ka, Anam-Dong, Sungbuk-Gu, Seoul, Korea
{hyuks00, sangjin}@korea.ac.kr
[2] Digicaps Co., Ltd., Jinjoo Bldg. 938-26 Bangbae-Dong, Seocho-Gu, Seoul, Korea
bskoh@digicaps.com

Abstract. In this paper, we design and implement the Flexible DRM (Digital Right Management) system based on using subject-mode for flexible multimedia service considering ubiquitous environment. The proposed system is complementary to inflexibility of the users which is a weakness of the traditional contents distribution system. Specifically, because a single using-subject can be flexibly changed under the ubiquitous environment, we attempted to seek the way to support the interactive transaction where both creation and consumption are available. In addition, It supports the super-distribution of the contents under the various environments. Furthermore, we design and apply the license format which can be flexibly used in multimedia devices under the wire or wireless environment.

1 Introduction

The rapid growth of computing power and the development of IT will be caused the Ubiquitous Environment (UE). It promises a computing infrastructure that seamlessly and ubiquitously assists users in accomplishing their tasks and that renders the using devices and invisible as embedding in device, user, and object. The key of ubiquitous is to deploy a wide variety of the intelligent devices at our living space [1-3].

In addition, At UE, the various digital data are converged including broadcasting, tele-communication, computer, and home electronic products are in combination. The basic aim of these converging UE is that it provides convenience and efficiency to users for their application such as home-automation, health-care, home-remote service, etc. UE results in change of everything such as things, environment, thinking, etc. [4-6]. Moreover, it needs to change of distribution mechanism at the contents distribution business field.

Previously, the traditional contents distribution architecture [7,8,19] was in fixed pattern, as the distribution using subjects - Contents Provider (CP), Contents Distributor (CD), and Contents Consumer (CC) - create, distribute, and consume the multimedia

* This research was supported by the MIC(Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Assessment).

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 236 – 244, 2006.

contents respectively, while in UE it would be necessary for each subject to have the distribution system that can be flexibly changed [20].

To protect and distribute multimedia contents in UE, the following requirements must be considered: contents protection and distribution, authentication and authorization, usage rule management and contents super-distribution, monitoring and events report, temper resistance, and interoperability [15-18].

We define DRM under the UE which contents can be actively transacted by flexible change of using-subject to "Flexible DRMS (Digital Right Management System)". In this paper, we designed Flexible DRMS and implemented its prototype. Furthermore, it is distribution system by supplementing and expanding MPEG-21 DID [9,10], IPMP [11,12], REL [13], and the mechanism of OMA-DRM [14]. Moreover, it is possible for CP to complete the appropriate authentication in UE, generate the contents, and then register the packaged contents in safety for its distribution. In addition, it is the Flexible DRMS where one can simultaneously consume the contents that the users from the different home network environments have generated and registered. Furthermore, we have designed and applied the license format which can be flexibly used in multimedia devices under the wire or wireless environment.

The rest of this paper is organized as follows. In Section 2, we describe design, implementation, and analysis of the proposed system. We conclude in Section 3.

2 Flexible DRMS

2.1 Flexible DRMS Design

The proposed system is based on [20], and extended and improved for UE. It can be divided into the interactive contents transaction in a single-domain and the contents transaction supporting the super-distribution among the multi-domains.

Figure 1 shows architecture of Flexible DRMS. In a single domain, various Flexible DRMSs are expressed by 1, 2, … n, and the expression in brackets means the mode of each subject that can be operated as CP, CD, CC, and so on.

Flexible DRMS in a single-domain was designed so that CPF mode and CCF mode for the supporting the interactive transaction of contents can be supported simultaneously for the flexibility of the distribution subject under the UE. In addition, Multimedia Streaming Server (MSS) is used for the contents streaming service among multi-domains. In the transaction of contents among multi-domains, Central Management Authority (CMA) takes the role to manage the overall services such as authentication, key management, license management, and so on. User parts of Flexible DRMS and the principle functions of each mode in CMA are as follows.

- ***CPF (Contents Providing Function) Mode*:** The copyright holder who provides the contents uses this CPF mode, and also takes charge of the CDF (Contents Distribution Function) mode function for its distribution. To take advantage of packager that is embedded in the Flexible DRMS, it creates digital items by adding the contents-related meta-data and the usage rule setting the usage rights.

  The digital item consists of the contents resource and the meta-data, being expressed in XML [12]. The CPF mode is designed to assist the super-distribution of contents, performing the super-distribution use packaging during the secondary distribution.

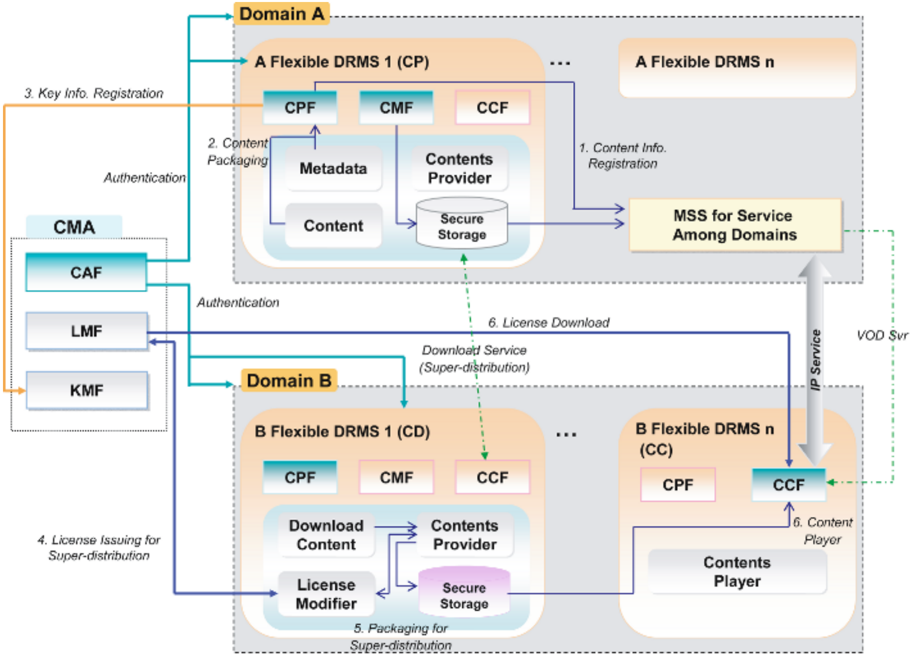


Fig. 1. Flexible DRMS Architecture

- ***CMF (Contents Management Function) Mode*****:** It manages the meta-data information that is created when it is packaged with the streaming server for the service of contents streaming within a single domain.
- ***CCF (Contents Consumption Function) Mode*****:** It is the mode for play of contents after receiving the issued license from CMA. License issuing procedure of CMA with respect to the LMF is as follows. When CCF requests the license from LMF, the license related to contents should be issued. When the license is requested, LMF is supposed to receive from CCF the Contents ID, Contents Header Information, public key of CCF and the user's information. Using the Contents ID, LMF takes from the Contents DB which is necessary to issue the license such as the rights information, meta-data information, and the key value and so on. Using the REL generator, it generates the license containing the rights information and the session key, and then transmits the license to CCF. LMF records the data about the issued license in the license issuing DB, and then records the information about the log with socket server.
- ***CMA*****:** It takes charge of the authentication about the subjects participating in the contents distribution, the issuing and management of the key used when packaging, and the issuing and management of the license to use the contents.
  · ***CAF (Certificate Authority Function) Mode*****:** It performs authentication of the domain and device, as well as the authentication for CP, CD, and CC which are the subjects participating in the contents distribution.

- ***LMF (License Management Function) Mode:*** It issues and manages the license that is used to decrypt the user's encrypted contents so as to play the packaged contents.
- ***KMF (Key Management Function) Mode*:** It issues and manages each key related to the symmetric key cipher algorithms used when packaging, and the public key cipher algorithms to treat the transaction of contents safely.

2.2 System Protocol

The proposed system can be divided into two services. One is for the interactive multimedia service in a single-domain, and the other is for the multimedia service to support the super-distribution among the multi-domains by using Flexible DRMS. The notations in the table 1 are used throughout this paper.

Table 1. Notations appeared in this paper

Notations	Meanings
Key, Pk, Sk, A_*k*	Symmetric key, Public key, Secret key, key *k* of A
C_k, D_k	128-bit key is used when encrypted/decrypted in the block cipher algorithms. (In reality, each key is the same key and conceptual notation)
$\|$	It concatenates the bit row consecutively
H( )	128 bit unique value using hash function
$E(\ )_{Pk}$	It encrypts into the public key cipher algorithms by using Pk
$D(\ )_{Sk}$	It decrypts into the public key cipher algorithms by using Sk
Rand	Value generated randomly
MD(A)	Meta-data for A
ConID	Contents ID
ConInfo / ConRight	Contents-related Information / Contents Rights Information
REL	REL-related Information such as usage period, usage frequency, etc.
DistInfo	Information on super-distribution or distribution
BankInfo	Bank Information to remit the created revenue amounts into the relative bank account
UID / PW	User ID / Password
X_y	X mode of y device or X device of y domain
Pack	Execution of Packaging
A*B	Execution A under B condition
A:B	Execution B by transmitting A: system subject, transmitting among the subjects.
A → B	Transmitting from A to B
DRMSxSN	Serial Number of DRMS's device

The following is the generation process for the key used in the proposed system.

(1) Key used when the contents are encrypted.
$Key=H(ConID\|DRMSxSN\|Ran[256]),\ x=1,\ 2,...\ n,\ \ D_k=Enc(Key)_{CMA\_Pk}$

(2) Key used when the LMF in CMA generates the license.
$Key=Dec(D_k)_{CMA\_Sk},\ Key1=H(D_k),\ C_{k1}=Enc(key)_{k1},$
$C_{k2}=Enc(C_{k1})_{DRMSx\_Pk},\ k=1,\ 2,\ ...\ n$

(3) Key used when the contents are decrypted
$key1=H(D_k)$, $C_{k1}=Dec(C_{k2})_{DRMSx\_Sk}$, $x=1, 2, \ldots n$

In the proposed system, we assume that the keys were pre-shared.

2.2.1 Interactive Multimedia Service in a Single-Domain

In this sub-section, we discuss interactive contents transaction in a single-domain. Moreover, we discuss in detail protocol among CPF of DRMS1, CMF of DRMS1, CCF of DRMS1, and LMF in CMA.

<CMA-Administrator : Authentication, Key Management, License Management>

(1) It authenticates CP, CD, CC, and DRMS device which are the subjects participating in the contents distribution.
CAF_{CMA} : $Enc(CPF||CMF||CCF)_{CMA\_SK}$, $H(key)$

(2) It issues and manages each key used in the symmetric key cipher system for packaging, and the public key cipher system for safe communication session, using the KMF in CMA.
$LMF_{CMA} \rightarrow CPF_{DRMS1}$: $Enc(UID||PW||CP_{Pk}||CP_{Sk})_{DRMS1\_Pk}$

(3) It issues and manages the license for playing the contents of CC.
$LMF_{CMA} \rightarrow CCF_{DRMSx}$: $License(ConName||ConRight||DistInfo||key||BankInfo)$,
$x=1, 2, \ldots , n$

<DRMS 1- CP, CD Mode: Contents Packaging and Registration>

(1) The copyrights holder of the contents, by using the CP mode of DRMS, inputs the contents-related meta-data such as contents category, kind, title, explanation, and URL.
CPF_{DRMS1} : $MD_{ConInfo}(category, kind, title, explanation, URL)$

(2) DRMS1 stores CMF with the inputted meta-data of the contents.
$CPF_{DRMS1} \rightarrow CMF_{DRMS1}$: $MD(ConInfo)$

(3) DRMS1 packages the contents, inputs the meta-data of rights using REL, and then registers on LMF in CMA.
CPF_{DRMS1} : $Pack(Content, MD(ConInfo, REL))$, $CPF_{DRMS1} \rightarrow LMF_{CMA}$: $MD(REL)$

<DRMS 2- CC Mode: Contents Consumption>

(1~2) CC downloads the contents list from CMF of the DRMS1. The follows is process of download : Contents lists can be used from TCP or HTTP. Contents list file format is delivered by XML format. CC selects the created contents out of the CMF of DRMS1.
$CMF_{DRMS1} \rightarrow CCF_{DRMS2}$: $ContentList(Download)$

(3) After CC's request of the license about the selected contents to LMF in CMA, rights authorization can be obtained from the LMF.
$CCF_{DRMS2} \rightarrow LMF_{CMA}$: $Licence(Request)$,
$LMF_{CMA} \rightarrow CCF_{DRMS2}$: $Licence(Response)$

(4) By requesting the streaming service from CMF of DRMS1, contents can be played.
$CMF_{DRMS1} \rightarrow CCF_{DRMS2}$: $Playing(Streaming)_{license}$

2.2.2 Multimedia Service Among Multi-domains

In this subsection, we discuss multimedia contents transaction among multi-domains. Furthermore, we discuss in detail protocol among CMF of DRMS1 at domain A, CDF or CMF of DRMS1, CCF of DRMS2 at domain B, and LMF in CMA.

<Domain A : DRMS 1- CP Mode: Contents Packaging and Registration>

The same as subsection 2.2.1

<CMA-Administrator : Authentication, Key Management, License Management>

The same as subsection 2.2.1 except it authenticates each domain.

<Domain B : DRMS 1- CD Mode: Contents Re-packaging and REL Modification>

(1) For the consumption of contents where the super-distribution among multi-domains is available, DRMS1 downloads the contents as CD mode through the CMF in domain A.
CDF_{DRMS1} *:Re-Pack(Packed Content, MD(ConInfo, Modified-REL))*

(2) After CD receives license from LMF in CMA, REL can be modified for super-distribution and then registered on the LMF in CMA.
$CDF_{DRMS1} \rightarrow LMF_{CMA}$*: MD(Modified-REL)*

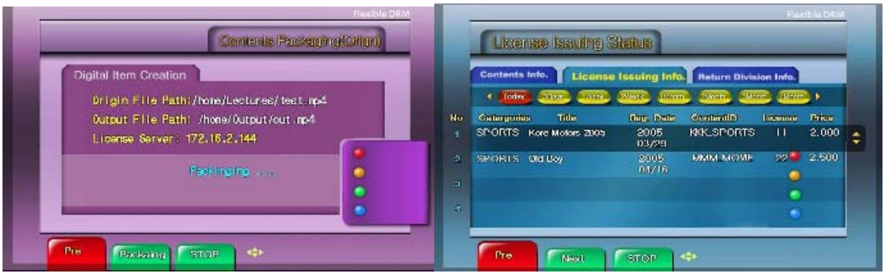
<Domain B : DRMS 2- CC Mode: Contents Consumption>

The same as subsection 2.2.1

2.3 System Implementation and Analysis

Figure 2 shows CP and CD mode in Flexible DRMS. The Left shows the process of contents packaging, after the meta-data relating the contents is inputted by CP as CPF mode of Flexible DRMS. The Right shows the issuing status of the licenses related with registration through LMF in CMA, after the REL related information with respect to the sales and usage regulation on the contents has been inputted.

The flexible REL format in the proposed system is supplemented and expanded distribution mechanism of MPEG-21, super-distribution and domain mechanism of OMA-DRM.



Origin File Path: /home/Lectures/ test.mp4
Output File Path: /home/Output/out.mp4
License Server: 172.16.2.144

No	Categories	Title	Reg. Date	ContentID	License	Price
1	SPORTS	Kore Motors 2005	2005 03/29	KKK_SPORTS	11	2.000
2	SPORTS	Old Boy	2005 04/16	MMM_MOVIE	22	2.500
3						
4						

Fig. 2. Screen of the contents packaging and the License Issuing Status

Table 2 shows the comparative analysis between the existing DRMS and the proposed system. The interactive transaction service means the end user can easily change from the existing mode into the other mode (CP, CD, CC) in a device. The proposed system supports such flexible change so that the interactive transaction could be possible. In addition, it supports the contents super-distribution among the multi-domains by which the CC who is good at the market characteristics of a certain domain can transact for the secondary distribution as a CP. Furthermore, in the proposed system, a flexible license format is supported between the fixed device and the mobile device, while in the existing system, license format was dependent on the devices. Key parts for encryption consists of key in KMF and the encrypted contents file, which are safely managed. It increases the efficiency of management by keeping the contents and meta-data independently. In addition, it supports download and streaming service, and therefore enables the creation of the various business models based on the interactive transaction.

Table 2. Comparison between the existing DRMS and the proposed system

Item Comparative System	Interactive transaction service	Contents super-distribution	Flexible License format support	Encryption key management	DI type	Service type
MS-DRM	X	O	□	Svr_key+ key for encryption	encrypt+ meta-data	Down / Streaming
Intertrust	X	X	X	Svr_key	encrypt+ license	Down
Proposed system	O	O	O	Svr_key+ key for encryption	encrypt+ meta-data +license	Down / Streaming

O : Excellent □: Normal X : Unsatisfied

The proposed system strictly discriminates the contents usage rights and the copyrights. Therefore, although CP transmits the contents through the unsafe communication channel, the usage of contents can be secured in confidence. However, the security of the user's terminal could be variable according to the usage environment of the contents. Therefore, the additional safeguard for the fixed or mobile terminal is necessary. The following is the potential attack that is likely to be made from outside, and the analysis of safety against those attacks.

- ***Protection from device spoofing attack:*** The proposed system can prevent the device spoofing attack by realizing the safe communication channel, using the public key cipher method for obtaining the license and authentication of the Flexible DRMS device as shown in the authentication of the subsection 2.2.1
- ***Protection from illegal license alteration attack:*** The proposed system can prevent the illegal license alteration attack by performing the integrity checking, including the digest value on the digital resources within the license.

- ***Protection from illegal user attack against contents:*** By the user authentication and device authorization through the certificate issued by CMA, illegal user's attack against contents can be prevented, and the non-reputation service can be also available.

3 Conclusion

In this paper, we have designed and implemented the Flexible DRMS which is suitable for UE. We have complemented weakness of the inflexible contents distribution among fixed subjects in the existing system. To take into account the environment where a single user can be flexibly changed in UE, it supports dynamically interactive contents transaction within an identical domain. In addition, it supports contents super-distribution among multi-domains that CC well known the market characteristic of the specific domain, can be easily changed and transact contents for the 2nd distribution. Furthermore, it supports license format which can be flexibly used in multimedia devices under the wire or wireless environment.

References

1. M. Esler, J. Hightower, T. Anderson, and G. Borriello. Next century challenges: Data centric networking for invisible computing. In Proceedings of the 5th ACM/IEEE International Conference on Mobile Computing and Networking (1999)
2. Mark Weiser: Hot topic: Ubiquitous Computing IEEE Computer (1993), 71-72
3. Mark Weiser: The computer for the 21 century. Scientific American (1991), 94–104
4. John Thackara: The design challenge of ubiquitous, Interactions, Volume 8, Issue 3, ACM Press (2001), 46-52
5. Jong Hyuk Park, Heung-Soo Park, Sangjin Lee, Jun Choi, Deok-Gyu Lee: Intelligent Multimedia Service System Based on Context Awareness in Smart Home. KES'05, Springer-LNAI, Volume 3681 (2005), 1146-1152
6. Anand Tripathi, Tanvir Ahmed, Devdatta Kulkarni, Richa Kumar, and Komal Kashiramka: Context-Based Secure Resource Access in Ubiquitous Environments, 2nd IEEE Annual Conference on Ubiquitous and Communications Workshops (2004)
7. Joshua Duhl, Susan Kevorkian: Understanding DRM Systems, IDC White Paper (2001)
8. William Rosenblatt, William Trippe, Stephen Mooney: Digital Rights Management: Business and Technology, Paperback (2001)
9. J. Bormans, K. Hill : MPEG-21 Overview v.5, ISO/IEC JTC1/SC29/WG11/N5231, International Organisation for Standardization, Shanghai (2002)
10. V, Iverson: MPEG-21 Digital Item Declaration FDIS, ISO/IEC JTC1/SC29/WG11/N4831, International Organisation for Standardization, Fairfax (2002)
11. Draft requirements for MPEG-21 DRM, ISO/IEC JTC1/SC29/WG11 N6271, Hawaii MPEG Meeting (2003)
12. MPEG DRM Extensions Overview, ISO/IEC W6338, MPEG Munchen Meeting (2004)
13. MPEG-21 Part 5: Right Expression Language FDIS, ISO/IEC JTC1/ SC29/WG11 N5599 (2003)
14. http://www.openmobilealliance.org, OMA-DRM-REQ-v2_0-20030515-C.pdf (2003)
15. Qiong Liu, Reihaneh Safavi-Naini and Nicholas Paul Sheppard.: Digital rights management for content distribution, ISSN:1445-1336, AISW2003(2003), 49 - 58

16. http://www.openmobilealliance.org, OMA-DRM-REQ-v2_0-20030515-C.pdf (2003)
17. DMP, TIRAMISU IST-2003-506983 DRM Requirements (2004)
18. ODRL Initiative Working Draft, Open Digital Rights Language (ODRL) Version 2 Requirements (2005)
19. IMPRIMATUR Web site: http://www.imprimatur.alcs.co.uk/html/home.htm
20. Jong Hyuk Park, Sangjin Lee, Kim Yeog, and Byoung-Soo Koh: Design and implementation of the IMS-IPMP System in Convergence Home-network Environment, ICADL 2005, Springer-LNCS, Volume 3815 (2005), 465 – 466

User Centric Intelligent IPMPS in Ubi-Home*

Jong Hyuk Park[1], Jungsuk Song[1], Sangjin Lee[2], Byoung-Soo Koh[3],
and In-Hwa Hong[4]

[1] R&D Institute, Hanwha S&C Co., Ltd., Jangyo-Dong, Jung-Gu, Seoul, Korea
{hyuks00, songjs}@hanwha.co.kr
[2] CIST, Korea University, 5-Ka, Anam-Dong, Sungbuk-Gu, Seoul, Korea
sangjin@korea.ac.kr
[3] DigiCAPS Co., Ltd., Jinjoo Bldg. 938-26 Bangbae-Dong, Seocho-Gu, Seoul, Korea
bskoh@digicaps.com
[4] KETI, Yatap-Dong 68, Boondang-Gu, Seongnam-Si, Kyunggi-Do, Korea
hongih@keti.re.kr

Abstract. In this paper, we design and implement the *intelligent IPMPS* (Intellectual Property Management and Protection System) in Ubi-Home. The proposed system supports flexible distribution platform for secure multimedia Service. In addition, we design user location recognition algorithm in order to provide intelligent services, and implement sensor network module using the algorithm to collaborate among devices in Ubi-Home. Furthermore, the proposed system provides multimedia service to authorized users who are using PC, STB, PDA, and Portable Device, etc. in Ubi-Home. Finally, Finally, we adopt the concept of domain authentication to improve the efficiency of license management for all device in Ubi-Home.

1 Introduction

Evolution of information technology has finally caused the up rise of the Ubiquitous Computing Environments (UCE). UCE aim to provide services of computer applications, embedded software and networked services in a highly flexible but integrated manner to users [1, 3]. It was advocated by Mark Weiser of Xerox PARC in 1998, and has provided the users with valuable services by interaction among numerous computers without user recognition. In addition, it has the ability to give any objects or space in daily life a sense of intelligence any time and any where [1, 2]. UCE are collaborative space including users, systems, services, sensors, and resources [4]. Furthermore, it results in affluent and convenient life at home. For this, it is necessary to develop intelligent system which can support user centric service among home appliances such as D-TV, PC, notebook, refrigerator, washing machine, microwave, etc.

Taking into account the aspect of multimedia service, there are several different frameworks and elements to distribute and consume multimedia contents in the

* This research was supported by the MOCIE (Ministry of Commerce, Industry and Energy)-HIMS Project (10016508) and the MIC (Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Assessment).

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 245 – 254, 2006.

current multimedia service environment. Nonetheless, there are no frameworks where the multimedia content can be collaboratively consumed and distributed among the different elements in home environment. Therefore, it is necessary to develop a systematic and efficient system to protect and manage the multimedia contents so that intelligent and secure multimedia services suitable for Ubi-Home can be available.

The following requirements need to be considered for multimedia service in Ubi-Home: user location recognition, low-power wireless network, flexible authentication of user and device, interoperability, contents protection and distribution, usage rule management, and contents super-distribution [5, 6, 7].

In this paper, we design and implement multimedia service system for the intelligent and secure service in Ubi-Home (U*i*IPMPS: Ubiquitous intelligent Intellectual Property Management & Protection System). The proposed system supports flexible distribution platform for secure multimedia service. Moreover, we design user location recognition algorithm in order to provide intelligent services and implement sensor network module by using the algorithm to collaborate among the appliances in Ubi-Home.

The rest of this paper is organized as follows. In Section 2, we discuss related works which are core technologies and research trends of multimedia service in home. In Section 3, we describe design included architecture and protocol, experiment, and analysis of the proposed system. We conclude and discuss the future research direction in Section 4.

2 Related Works

In this section, we describe core technologies of ubiquitous computing implementing Ubi-Home, and then outline research trends of DRM in Home-network.

Wireless Sensor network (WSN) is a core technology implementing Ubi-Home environments. It is consisted of sensor node. Densely located micro controller is embedded into sensor network. Sensor node have sensing module, data processing module and communication module inside [8].

Context Awareness (CA) is a technology allowing communication between user and computer. Its aim is to bring the standard of understanding of communication up to such level of human world. CA recognizes the situation (a location, a place, sound levels, duties, private situations and time) of the user being faced and acquires correctly the information in accordance with the situation as a desired form. The system which provides above resources is considered as a context awareness system. The situations mean situation information for at least one object. It is a person, a place, time and subjects that could be the object of situations and they are suitable factor between users or applications. The computing resources equipped the ability of context awareness are required the function that obtains and extracts a situation data, converts the data into the form suited for present situation [9, 10].

We outline some research trends of secure multimedia service in Home-network. Recently, Digital Video Broadcast and TV Anytime have turned their attention to content protection on the emerging Home-network. There are at least three proposals on the table; Thomson' s SmartRight, Cisco' s OCCAM, and IBM' s xCP Cluster Protocol. Thomson's SmartRight is based on smart cards in every device [11].

Other standards initiatives related to Home-network began to gain momentum in 2004. The 4C and 5C consumer electronics consortia, contents media storage standards, have been defining the standards related to content storage media CPRM/CPPM (Copy Protection for Recordable/Prerecorded Media) and the network protocols for inter-device communication-DTCP (Digital Transmission Content Protection) respectively. Toshiba, one of the members of the 4C Entity, introduced CPRM-compliant DVDs this past year. These specs are located well, because they apply to what everyone agrees are the essential building blocks of home digital entertainment networks, not to home entertainment network "solutions", whatever that means [12].

3 U*i*IPMPS

3.1 U*i*IPMPS Design

In this paper, the proposed system was designed considering multimedia contents in various environments such as extra-VoDs, Terrestrial / Satellite DMB (Digital Multimedia Broadcast), Cable TV, etc. In addition, its main components consists of three elements such as follows:

- ***i*MG:** It receives and stores digital contents from a extra CP (Contents Provider), and trans-codes to provide contents flexiblly among appliances in Ubi-home. In addition, It takes access control functionality for providing multimedia contents in various devices at room 1, room 2, room 3, and so on.
- **WSN module:** It takes functionality for network communication among WSN modules, and context collection functions for each object attached on end user and IMPD. WSN module consists of low power manager, sensor part, operating system, and the protocol module, middleware part, and application program.
- **U*i*IPMPS Client:** It takes basically functionality to depackage packaged contents and update right informaion about contents access through the license parsing. Furthremore, it takes functionality to display multimedia such as DTV, PDA, mobile note book, etc., and embeds WSN module, so multimedia device context information is delivered to *i*MG through WSN.

Figure 1 shows architecture of U*i*IPMPS. It is divided into user location recognition part and secure multimedia service part. The former support user centric intelligent service. The latter consists of VOD supporting MPEG-2/4, multimedia protection, and service providing real-time streaming service.

3.1.1 User Location Recognition Part in Ubi-Home

Wireless network randomly assign the nodes and voluntarily have to transmit the information corresponding surroundings. To achieve the active information transmission, trusted location information and data transmission is very important. Previous wireless network doesn’t support the functionality to transmit the information corresponding location recognition and active surroundings.

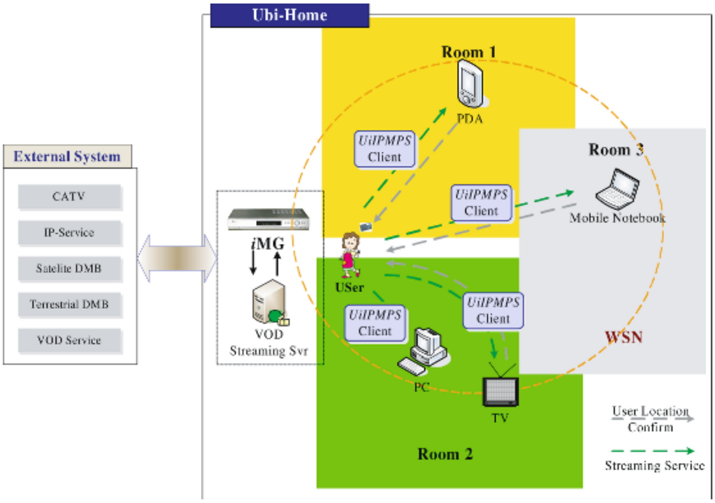


Fig. 1. U*i*IPMPS Architecture

In this paper, we propose the recognition technology detecting the location of the moving node. The node will be randomly assigned after measuring the RSSI (Received Signal Strength Indicator) based on the grid typed wireless network. Moreover, our proposed user location recognition algorithm consists of two parts. One of the two consisting parts are triangular measuring using RSSI measurement and RSSI and the other is a distance measurement by triangular measurement and average speed of moving node. User location recognition algorithm follow steps: RSSI sampling → Location calculation → Error compensation → Estimation. In this process, triangular measuring method has a advantage that we can calculate the distance using RSSI relative formula depending distance. If we know the distance from node, we can get the relatively exact location after the minimal operation using triangular measurement. To measure the RSSI, it needs a message formatting process being consisted of *RSSI Request Message* and *RSS Reply Message*. Therefore, we apply Tiny OS's MAC [13] developed by the UC Berkley.

3.1.2 Secure Multimedia Service Part in Ubi-Home

The secure multimedia service part is divided into streaming service transmission and receipt part. Transmission part including streaming server, license server, content packager, and content management server. Streaming server provides VOD and Multicast live streaming service. License server performs general right protection throughout the digital content distribution process. Content management server provides statistic and monitoring service for the log of content license.

Receipt part by *i*MG (intelligent Multimedia Gateway) of devices and this intelligent home devices are assigned to one of the domain. That's, the user requiring device is contained in individual domain. The contained devices make the share and use of content possible.

- ***UiIPMPS Domain:*** The domain might be considered as the set of devices which sharing the same content and license. To manage the free sharing and distribution of the content and license among devices in Ubi-Home. We extanded and applied the domain concept of OMA-DRM [6] into this system.

In U*i*IPMP Domain, *i*MG takes function of the domain controller and all devices in domain share the private domain key. In addition, One domain has one key at least and one device can be registered in several domains. The Scenario for registration and secession is depicted in Figure 2. There are four devices registered in domain-00. Even if device 3 and 4 are seceded, they still having a domain key. Therefore, they can use the pre-purchased contents. But they can't share a license of the new content. Devices in domain-01 have the key for both domain-00 and domain-01, so that they can use all contents corresponding to these domains. Moving among domains depends on the license policy for each content. As shown the Figure 2, Device 3 registered in domain-00 is impossible to use the content contained device 1 in domain-01. But device 2 registered in domain 01 can use the contents purchased contents in domain-01 and also can use the contents distributed from domain-00.

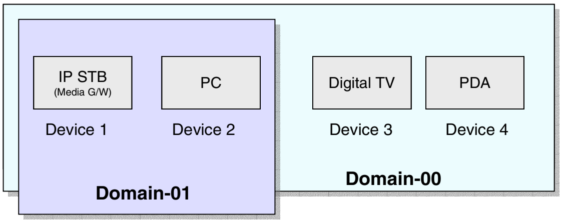


Fig. 2. Domain registration / secession of the device

- ***User Identity Monitoring:*** The understanding user's identity is very important to provide customized service for each user. There are several ways to understand the recent user. The more effective way is to store the user's ID into sensor network in advance and use the content. We assume that the user using the terminal is unique, understanding that what type of terminal is used by certain is a different technical problem. We differentiate the user by restoring the user's English name between <user> and </user> tag of XML format. Moreover, The media parameter using this type of tag.

3.2 Proposed Protocol

The notations in table 1 are used throughout this paper.

In this paper, the proposed system consists of license issuing protocol and delivery protocol between *i*MG and device (or client).

<License Issuing Protocol>

In the proposed system, license is defined with those components. Fundamental license is composed of the follow:

$$License=(ContentID,\ DeviceID, K_{PACK},\ Certificate_{license})$$

Table 1. Notations appeared in this paper

Notations	Meanings
ContentID	Unique value of content for multimedia service
UserID	Registered unique ID in *i*MG
DeviceID	Unique value for device identification
Pac_{Key}	Key used for multimedia content packaging
License	License with electronic signature
AUS	Certificate Server to authenticate user, device, and domain
iMG	Intelligent Multimedia Gateway in Ubi-Home
CMS	Contents Management Server
VSS	VOD Streaming Server
$Certificate_{X.509v3}$	Certificate in X.509v3 format (Simple Certificate)
$Certificate_{user}$	User Certificate
A	Content 'A' for multimedia service
E(A)	Content 'A' packaged
DRM_{Client}	Depackaging content and identifying the authority of user or device for content
Auth( )	User/Device Authenticate
H( )	Hash Function
Cert( )	Certificate Data

ContentID is a unique authority code of distributing content. It represents the authority of digital content apparently and can be used as a common query key for each content. DeviceID is a user's special DeviceID which is created to bind user with hardware. In this paper, inherent value for device and UserID is stored in the DeviceID. This DeviceID is iteratively creating the sequence number to keep secure streaming service and re-constructed value is used to convert into legal stream data in the client. Rights receive variable level due to the importance of content. Every level has a limitation of available number, period, and the number of device, etc. The general device information is as a follow;

$$Device\_Info = H(UserID \| DeviceID)$$

User gets the right to use the content by acquiring license. If user uses player, management module of client will protect illegal use of license or content by transmitting the user information to Multimedia Service Server. The following steps are used for license request and acquisition for multimedia service:

<u>*Step 1.*</u> User requests license issuing to AUS to get the right of content.
$User \rightarrow AUS{:}Data(UserID \| DeviceID \| ContentID)_{Request}$

<u>*Step 2.*</u> AUS identifies the userID and DeviceID.
$AUS \rightarrow User{:}H(ContentID \| License \| Certificate_{X.509v3})$

<u>*Step 3*</u>. AUS encrypts license by user's public key and transmit to client.
$AUS \rightarrow Auth(UserID, DeviceID)$

<u>*Step 4.*</u> USer send the employment list into CMS after receiving the license.
$User \rightarrow CMS{:}License(UserID \| ContentID \| Certificate_{user})$

<Part 1>

1. Device transmits its own UserID to *i*MG, and requests a streaming service of content ‘A’.
   $Device \rightarrow iMG{:}A_{Request}\,(UserID||DeviceID||ContentID)$
2. *i*MG transmits to CMS the information received from device and requests a streaming service of content ‘A’.
   $iMG \rightarrow CMS{:}A_{Request}\,(UserID||DeviceID||ContentID)_{Request}$
3. CMS requests authentication from AUS.
   $CMS \rightarrow AUS{:}Auth_{Request}\,(UserID||DeviceID||ContentID)$
4. AUS transmits authentication information to CMS.
   $AUS \rightarrow CMS{:}Cert(DeviceID, UserID, Pac_{Key}, ...)$
5. CMS receives authentication information and requests from VSS, a streaming service of packaged content ‘A’.
   $CMS \rightarrow VSS{:}Request_{E(A)}$
6. VSS performs a streaming service of packaged content ‘A’ and *i*MG receives it.
   $VSS \rightarrow iMG{:}Streaming_{E(A)}$

<Part 2>

7. *i*MG requests user location information to WSN module.
   $iMG \rightarrow WSN\ module{:}Request_{Msg}$
8. WSN module transmits response packet about user location information to *i*MG.
   $WSN\ module \rightarrow iMG{:}Reply_{Packet}$
9. *i*MG transmits packaged content ‘A’ to U*i*IPMPS client in order to depackage content ‘A’ based on user location information.
   $iMG \rightarrow UiIPMPS_{Client}{:}E(A)$
10. U*i*IPMPS client depackages content ‘A’ and provides multimedia streaming service according to the type of devices (D-TV, PDA, Portable device, etc.).
   $UiIPMPS_{Client} \rightarrow Device{:}A_{Content}(Streaming)$

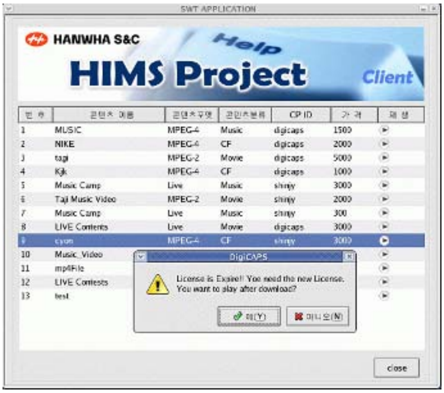
3.3 Experiment and Analysis of the U*i*IPMPS

In this subsection, we discuss experiment and analysis of the proposed system. Figure 3 shows the screen that license wad checked before the contents are playing.

Table 2 is a part of REL to represent license right on proposed flexible Ubi-Home. It extends the Super-distribution and Domain of OMA DRM. Distribution mechanism for MPEG-21 REL.

We compare the existing system with the proposed U*i*IPMPS. The existing system means multimedia service at home network systems of K-Com.[14] or S-Com.[15] demonstration enterprise consortium in Korea.

The proposed system makes the authentication of a certain device or specialized device possible and support flexible license depends on the device ID as shown table 2. Even existing license format is dependent on device, the proposed system support the flexible license format. This license is mutually interoperable between fixed device and mobile device. As it authenticates the device by applying the domain and X.509 device certificate, free sharing and distribution of digital content is available in devices assigned to domain.



번 호	콘텐츠 이름	콘텐츠구분	콘텐츠분류	CP ID	가 격	재 생
1	MUSIC	MPEG-4	Music	digicaps	1500	▶
2	NIKE	MPEG-4	CF	digicaps	2000	▶
3	tagi	MPEG-2	Movie	digicaps	5000	▶
4	Kjk	MPEG-4	CF	digicaps	1000	▶
5	Music Camp	Live	Music	shinjy	3000	▶
6	Taji Music Video	MPEG-2	Movie	shinjy	2000	▶
7	Music Camp	Live	Music	shinjy	300	▶
8	LIVE Contents	Live	Movie	digicaps	3000	▶
9	cyon	MPEG-4	CF	shinjy	3000	▶
10	Music_Video					▶
11	mp4File					▶
12	LIVE Contests					▶
13	test					▶

DigiCAPS

License is Expire!! You need the new License.
You want to play after download?

Fig. 3. License checking

Table 2. Parts of flexible REL format in *i*MS-DRM

```
// field where multimedia parameter is handled for location confirmation.
<Nodes>
   <Node id="6400"><BaseId>7e00</BaseId>
      <Computing><Codec>mpeg4,mpeg2</Codec><Network>wlan,bt</Network>
      <IP>192.168.1.10</IP><Resolution>1024x768</Resolution>
      </Computing><Foraging>
         <Genre>movie,drama,sports,news,documentary</Genre>
         <Keyword>ring,band,brothers</Keyword>
        <Device>tv,radio,pc,pda,fridge</Device></Foraging></Node>
</Nodes>
// Domain structure for flexible content and device certification
   <sx:dnsName>urn:domain Name:device or user id</sx:dnsName>
   <sx:commonName>urn:STB:Serial Number</sx:commonName>
// Granting certification key value on content ID concerned.
<asset><context><uid> cid:12345 </uid> </context><KeyInfo>
       <KeyValue>vUEwR8LzEJoeiC+dgT1mgg==</KeyValue>
   </KeyInfo>
  </asset>
// Restricting frequency and period for granting authority of media service.
<DigestValue>WgCxegWxrpb2kBOSttmf2P8ZFLI=</DigestValue>
<SignatureValue>Ir+YdkpisfOvAIyLR+emQu9UHmnnnQ/bNQ=….
</SignatureValue>
// limit counts and duration for authorization of multimedia service.
<permission>
   <play><constraint><count>5</count></constraint></play>  ………………
```

In the existing system, user selects preferred contents after streaming service of contents provided by CP at home, and commands manually by selecting the device to play. However, the proposed system provides intelligent user-centric multimedia service suitable for Home enabling automatic execution of information without the passive intervention of the user by using WSN module to collaborate in among devices in Ubi-Home. The existing system is operated by XrML based on license system for the safety of general online contents in user rights management. This system has an inconvenience of user to reconnect to the main license server through external line in case of switching of user rights between user A and user B. Whereas, in the proposed system, the convenience of user and the efficiency of user rights management can be increased by using a license system capable of switching user rights.

The followings explain the security of the proposed system against possible attack in Ubi-Home.

- ***Protection from license modification and copy:*** By managing authority of user and contents through AUS for a certain period of time, we can verify the license and then guarantee the integrity from the attacks such as a handling of the license number and change of authority. After DigestValue is encoded by Base64 using private key for packaging, we encrypt above value using Base64 and then utilize as a signature as shown table 2 in subsection 3.3. In client parts, the Digest Value is verified first followed by the verification of the Signature value. They can confirm whether the license is modified or not through the two steps.
- ***Protection from illegal eavesdropping:*** Even if a eavesdropper copies device info. From device A to device B, the propose system can protect from illegal eavesdropping, because it is hashed by *H(UserID||DeviceID).*
- ***Protection from the malicious network attack:*** The proposed system guarantees confidential service from malicious attacks on network, because all packets in home network encrypted with symmetric key with 128/256 bits. The proposed system extended and applied ISMA (Internet Streaming Media Alliance) [16], secure streaming protocol, to provide service streaming service in Ubi-Home.
- ***Protection from the user disguise:*** The proposed system can protect from user disguise through authentication of the user, device, and domain by using X.509v3 Certificate. In addition, even if both user and device are authenticated by AUS, if they didn't register to domain as described subsection 3.1.2, user can't be provided multimedia service.

4 Conclusion

Our new system provides user centric intelligent secure multimedia service in Ubi-Home. The proposed system supports flexible distribution platform for Secure Multimedia Service. Moreover, we design user's location recognition algorithm (not available in previous work) in order to provide intelligent services, and implement sensor network module using the algorithm to collaborate among devices in Ubi-Home. Furthermore, the proposed system can provide multimedia streaming service to proper users which are using PC, STB, PDA, and portable device, etc. It adopts the concept of domain authentication to serve multimedia streaming service to legal user then it improves the efficiency of license management for all devices in Ubi-Home.

In the future, we should send more time to study optimized service among incompatible devices. The optimized service contemporarily serves suitable resolution with considering multimedia device status. We are going to graft the proposed system and model which is considering user's privacy together.

References

1. Mark Weiser: The Computer for the 21st Century, Scientific American (1991)
2. Mark Weiser: Hot topic: Ubiquitous Computing, IEEE Compute (1993)
3. K. Carey, D. Lewis, S. Higel, V. Wade: Adaptive Composite Service Plans for Ubiquitous Computing, in proc. 2nd International Workshop on Managing Ubiquitous Communications and Services (MUCS), Dublin (2004), 13-14
4. G. Zhang and M. Parashar, Context-aware dynamic access control for pervasive computing, CNDS'04, USA (2004)
5. DMP: TIRAMISU IST-2003-506983 DRM Requirements (2004)
6. http://www.openmobilealliance.org, OMA-DRM-REQ-v2_0-20030515-C.pdf (2003)
7. Qiong Liu, Reihaneh Safavi-Naini and Nicholas Paul Sheppard: Digital rights management for content distribution, AISW2003 (2003)
8. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” IEEE Communication Magazine, Volume 40, No. 8 (2002), 102-114.
9. A.K. Dey and G.D. Abowd: Towards an understanding of context and context awareness, HUC99 (1999)
10. Jong-Hyuk Park, Jun-Choi, Sang-Jin Lee, Hye-Ung Park, and Deok-Gyu Lee, “User-oriented Multimedia Service using Smart Sensor Agent Module in the Intelligent Home", CIS'05, Springer-LNAI, Volume 3801 (2005), 313 – 320
11. EICTA: Content Protection Technologies, http://www.eicta.org/copyrightlevies/index.html
12. Bill Rosenblatt: Year In Review: DRM Standards, http://www.drmwatch.com/standards, DRM Watch(2004), (2005)
13. Tiny OS Community Forum, http://www.tinyos.net
14. Korea Telecom Company Website, http://www.kt.co.kr
15. SK Telecom Company Website, http:// www.sktelecom.com
16. ISMA Website, http://www.isma.tv

The Design and Development of a Secure Keystroke System for u Business

Hangbae Chang[1], Kyung-Kyu Kim[2], Hosin Lee[3], and Jungduk Kim[4]

[1] SoftCamp Co., Ltd.,
828-7 Yeoksam Dong Gangnam Gu, Seoul, 135-080, Korea
hbchang@paran.co.kr
[2] Yonsei University,
134 Sinchon-Dong, Seodaemun-Gu, Seoul, 120-749, Korea
Kyu.kim@yonsei.ac.kr
[3] University of IOWA,
Iowa City, IA 52242, United States
skytrust@gmail.con
[4] Chung-Ang University,
72-1 Nae-ri Daedeok-myeon Anseong-si, Kyunggi-do, 456-756, Korea
jdkimsac@cau.ac.kr

Abstract. In combination with the easy acquisition of hacking tools and new artificial intelligent computer viruses such as Bug Bear, Spy Ware, or Net Devil, it demands a new information security area for the keyboard input information. In this study, considering the limit of the prior technologies, a source security method of the keyboard input information, regardless of the type of the hacking tool, was developed. The method makes use of a newly developed keyboard security driver at the Kernel level departing from the Pattern Matching Heuristic Method which requires technologies corresponding to various hacking tools case by case basis. In order to verify the technology developed in this research, tests were carried out in comparison with the prior arts with collected hacking tool, which have contributed to the development of the suggested technology.

1 Introduction

1.1 Research Background

Recently, there have been several banking theft cases by hackers who have secretly implanted hacking tools into the victims' computers which could extract the banking information entered through the victims' keyboard inputs. The information was transmitted through network and illegally used to draw the victims' bank deposits. This type of infringing personal information cannot be prevented with the information security technologies described earlier. In combination with the easy acquisition of hacking tools and new artificial intelligent computer viruses such as Bug Bear, Spy Ware, or Net Devil, it demands a new information security area for the keyboard input information.

As a secure method of keyboard input information, a new technology of entering personal sensitive information with mouse and screen keyboard input, in place of

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 255 – 261, 2006.

keyboards. However, the technology has not yet been matured enough to support the control of various operating systems and connection ports simultaneously securing the overall section of the transmission of the entered information.

In this study, considering the limit of the prior technologies, a source security method of the keyboard input information, regardless of the type of the hacking tool, was developed. The method makes use of a newly developed keyboard security driver at the Kernel level departing from the Pattern Matching Heuristic Method which requires technologies corresponding to various hacking tools case by case basis.

1.2 Research Methodology

In this research, the information security technology development methodology had been applied which develops technology to resolve the vulnerability problems in accordance with the analysis result of the vulnerability derived from the process of risk analysis(Peltier, 2001). On the basis of this methodology, the process is consisted of the identification of the information assets from the information flow entered through keyboards, the analysis of the security vulnerability of each steps, and establishment of the security measures with pertinent technology.

In particular, analyze the information flow entered through keyboards in Windows environment, and derive possible security vulnerability on the basis of the analysis. In order to resolve the derived security vulnerability, examine prior studies to identify their limitations, and design and develop the technology suggested in this research to resolve such limitations. Finally, the conclusions obtained from the experiments conducted with the technologies of prior studies and this research is presented together with the direction of future studies.

2 Vulnerability Analysis of the Process of the Keyboard Entered Information

The vulnerability in security which results in the leak of keyboard entered information can be exposed in various steps by various hacking tools, in the process of the keyboard input from the keyboard hardware to the application software.

At the keyboard input/output port, the keyboard scan code in the keyboard input/output port remaining after the transmission of the scan code from the CPU can be hacked by hacking tool(Port Scan). Common keyboard filter drivers have the functions of reading and controlling the communication of the keyboard input information, providing an environment which enables keyboard manufacturers to add functions to the PS/2 keyboard hardware. However, such keyboard filter driver cab be used as a tool to extract the keyboard inputs in the procedures of processing the input information. The most common method of extracting keyboard inputs is the Hooking technology which interrupts the function call from the application software and make the keyboard inputs processed with the function prepared and implanted by the hacker. In the process of the keyboard entered information (WM_KEYBOARD) in Windows environment, a hacker can interrupt (Event Dispatch) the keyboard inputs with the method described below.

3 Literature Review on the Security of the Keyboard Information

The methods of resolving the vulnerability problems which can be occurred in the process of the information entered through keyboard can be classified into hardware methods and software methods. In the hardware methods, in general, additional encrypting device transforms all the keyboard entered information. The software method secures keyboard inputs with application software including the Screen Keyboard which converts the keyboard inputs at the application software level and the keyboard Security Driver which is installed in the computer system to encrypt keyboard inputs.

In this section, the limits of the prior studies of the keyboard input security are identified by analyzing their implementation principles to apply for the development of the technology to resolve the problem.

3.1 Hardware Approaches

User enters desired information into the keyboard hardware, which is encrypted by additional devices. There are a diversity of keyboard input encrypting devices from the simple one which shifts the inputs using the Shift Register Circuit to the sophisticated one which converts the inputs with the combination of the time variable using the system clock signal and the Arithmetic Circuit (Michael F. Angelo, 1998, David Carrol Challenger, 2003).

The hardware methods require encryption software to be designed and installed into the hardware chip. Therefore, they are difficult to modify and expensive. Furthermore, this method can prevent information leak by keyboard input message hooking, but still vulnerable to other types of keyboard input process.

3.2 Software Approaches

There is a conventional method of indicating the user inputs, such as passwords, with predetermined same symbols (e.g., '*', '#' etc.), not with the characters entered by the user. However, keyboard inputs in this method can be leaked by combining the characters in the keyboard hardware area entered by the user or by analyzing the computer system storage which temporarily saves the keyboard inputs. To resolve such problems, software approaches have been studied to encrypt user keyboard inputs at diverse levels (user, system) without additional hardware device.

Among such approaches, the method which changes the keyboard input process according to the result of the inspection of the hacking tools makes combined use of the encryption and screen keyboard method. This method requires keyboard security driver installation, key logger decision, screen keyboard generator, and keyboard input encryptor(Ahn Lab, 2003). In this research, the problem of having to reboot the system with the keyboard security driver, which had been developed earlier, at the bottom level when it fails to preoccupy the inputs entered from the keyboard a the system level due to the order of loading device drivers.

However, this method is still vulnerable to security problems in the prior stage to the virtual keyboard driver in case that the keyboard inputs are encrypted, or after the virtual keyboard driver in case that the keyboard inputs are replaced with the screen keyboard inputs.

4 Suggested Technology of Securing Keyboard Input Information

In this research, a technology of securing keyboard input information at the system level by analyzing possible security vulnerabilities at the keyboard input procedure and the limits of the resolutions in the prior arts.

4.1 Design of the Keyboard Input Information Security System

The suggested system is consisted of the Secure Web Page Control installed in the server, the Debug Exception Processing installed in the security keyboard driver, Interrupt Vector Table Monitoring, and Keyboard Input Data Encryption.

The Secure Web Page Control carried out the functions including the environment inspection of the user's computer system, installation and notification of the keyboard security driver, checking the activation (focus) of the Web page, and decryption and output of the keyboard inputs. The keyboard security driver carries out the analysis of the keyboard input information, Debug Exception Processing, modification of interrupt vector table and redefinition of the process function, and encryption and transmission of the keyboard input information.

- Resolution of the Keyboard Input/Output Port Scan

As for the secure methods of the keyboard input information left in the keyboard input/output port, the method of deleting the information using the hardware control command and the method of controlling(blocking) the access to the keyboard input/output port are available. The former method controls the keyboard input/output port using separate hardware control command. On receiving the keyboard input information, the keyboard security driver sends the control command(Enable Keyboard, F4h) commonly defined in the keyboard hardware to the keyboard input port(60h - IN). On receiving the control command, the keyboard hardware activates(Enable, Reset) the keyboard and sends the acknowledge message to the keyboard output port(60h - OUT). In this process, the residual information in the keyboard output port is replaced with different information, which is of equal effect of deletion. This method cannot prevent the interruption of the information in the keyboard port by other hacking tool in the process of the deletion. Furthermore, the OS has to carry out the keyboard input process twice, which may result in the slower speed of the keyboard input process.

The access control method of the keyboard input/output port secures the keyboard input information by deleting the register wherein the keyboard input information is stored, when any hacking tool except the OS and keyboard security drive is detected, using the Debug Exception Processing supported by the OS, to access the keyboard input/output port.

- Encryption of the Keyboard Input Information

The process function of the keyboard interrupt analyzes whether the keyboard input is a common character key or a special function key(Ctrl, Enter, F1, Tab, etc.) to process accordingly. Character key inputs are encrypted using the Key Tale and special function keys inputs are processed by the system inherent interrupt process.

- Resolution of the Window Message Hooking

Keyboard input hooking by hacking tools is resolved by transmitting encrypted keyboard inputs directly from the keyboard security driver to the Web page security control to carry out the decryption at the keyboard input process step.

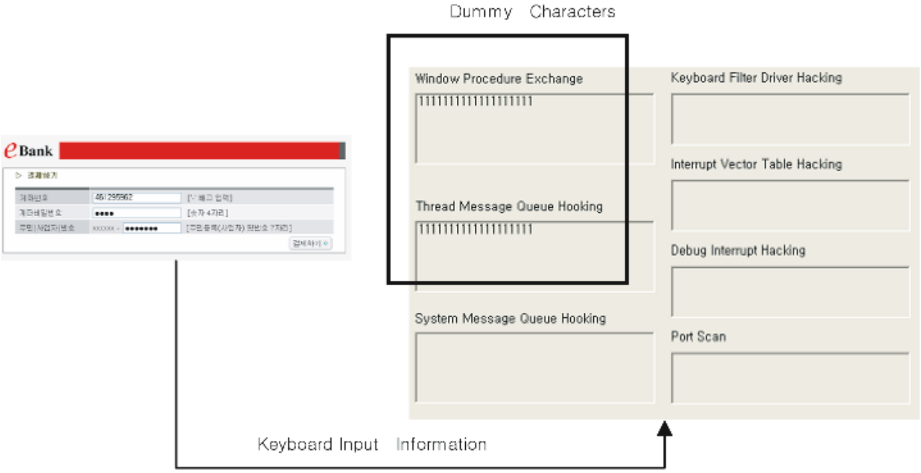
5 Research Methods

The security performance of the technology developed in this study was tested with the collected hacking tools and compared with the existing technologies to prove the advantages.

As for the major prior arts of the security of the keyboard input information, hardware method(David Carrol Challenger, 2003), Screen Keyboard method(Ahn Lab, 2003), and the encryption engaged into the keyboard security driver(Shakshuki, 2005) have been implemented and compared with the method developed in this study. Before proceeding the test, a hacking tool which attacks the security vulnerabilities at various stages simultaneously has been implemented for the test. The implemented hacking tool has user ID('gil-dong') and password('hong') from the keyboard port scan to the Windows message hooking entered by the user at each step.

The technology suggested in this paper has resolved the information leak by the keyboard input/output port scan with the Debug Exception Processing, and the information leak by filter driver installation and message hooking with direct transmission of the keyboard input information to the application software.

Fig. 1 shows that different from the prior arts wherein the keyboard input information can be leaked at various steps, the technology developed in this research can secure the information safely from the steps prior to the keyboard input information processing. From the keyboard input/output port to the interrupt vector



eBank

▷ 조회하기

계좌번호 461295962 ['-' 빼고 입력]

계좌비밀번호 •••• [숫자 4자리]

주민|사업자번호 xxxxxx - ••••••• [주민등록(사업자) 뒷번호 7자리]

결제하기

Window Procedure Exchange

1111111111111111111

Thread Message Queue Hooking

1111111111111111111

System Message Queue Hooking

Keyboard Filter Driver Hacking

Interrupt Vector Table Hacking

Debug Interrupt Hacking

Port Scan

Fig. 1. Result of the Suggested Technology Applied to the Hacking Tool

Table 1. Comparison Experiment to Prior Studies

	Hardware Method (David Carrol Challenger, 2003)	Screen Keyboard Method (Ahn Lab. 2003)	Encryption at the Keyboard Security Driver(Shakshuki, 2005)	Technology Suggested in this Research
Keyboard Input Message Hooking (Thread Message queue)	O	X	O	O
Keyboard Input Message Hooking (System Message Queue)	O	X	O	O
Filter Driver Installation	O		O	O
Keyboard Input/output Port Scan	X		C	O

※ O : keyboard inputs are secured from the hacking tools
X : keyboard inputs are not secured ed from the hacking tools

table, the access of hacking tools is prevented fundamentally. The encrypted keyboard input information is directly transmitted without passing through the filter driver. System message queue and the thread message queue show the transmitted dummy information for the keyboard input information received by the security Webpage control.

5.1 Research Results and Directions for Future Studies

In this research, the section from the keyboard input to the Web page or application software is secured with a newly developed technology. On the basis of the information security methodology, the process of identification of the information assets of the information flow of the keyboard input, analysis of the security vulnerability by steps, and the establishment of the technical resolution have been performed. To resolve the security vulnerabilities, the prior arts had made use of hardware or software methods, however, their functions and security were limited. Therefore, a keyboard input information security technology at the system level was developed in this research to cope with the limitations in the prior studies. In order to verify the technology developed in this research, tests were carried out in comparison with the prior arts with collected hacking tool, which have contributed to the development of the suggested technology.

This technology can be applied to all the e-business transactions which require personal information. It also can generate basic data for the investigation of keyboard input information leak, if it happens. It is expected that this technology can provide customers with confidence as well as preventing the accidents caused by leak of personal information which often occur in recent days.

References

[1] Andrew Ren Wei Fung, Cow Jean Farm, Abs C. Lin, "A Study on the Certification of the Information Security Management s Systems", Computer Standards & Interfaces, 2003.

[2] Wold, G. H. & Shriver, R. F., "Risk Analysis Techniques", Basic DR Articles, Disaster Recovery Journal, 1997.

[3] David W. Biessener, and Gaston R. Biessener, "Virtual Physical Drivers", US Patent, 0204700, 2003.

[4] Michael F. Angelo, "Method and Apparatus For Providing Secure and Private Keyboard Communications in Computer Systems", US Patent, 5748888, 1998.

[5] Helen Custer, "Inside Windows NT", Microsoft Press, 2003.

[6] David Carroll Challenger, "Apparatus and Method for Verifying Keystrokes within a Computing System", US Patent, 6630926, 2003.

[7] Rykut Guven, Ibrahim Sogukpinar, "Understanding Users Keystroke Patterns for Computer Access Security", Computer & Security, Vol. 22, No. 8, 2003.

[8] Saleh Bleha, Charles SLIVINSKY, Bassam Hussien, "Computer Access Security Systems Using Keystroke Dynamics", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 12, No. 12, 1990.

[9] AhnLab, ”Method for Protecting from Keystroke Logging”, Korea Patent, 10-0496462, 2003.

[10] Shakshuki Elhadi, Luo Zhonghai, Gong Jing. "An Agent-Based Approach to Security Service", Journal of Network & Computer Applications, Vol. 28, Issue 3, 2005.

Linkability of a Blind Signature Scheme and Its Improved Scheme

Jianhong Zhang[1,2], Tao Wei[1], JianYu Zhang[1], and Wei Zou[1]

[1] Institute of Computer Science & Technology, Peking University,
Beijing 100871, P.R.China
{zhangjianhong, zouwei}@icst.pku.edu.cn
http://www.icst.pku.edu.cn

[2] College of Science, North China University of Technology,
Shijingshan District, Beijing 100041, P.R.China

Abstract. Blind signature allows a user to obtain signatures from an authority on any document, in such a way that the authority learns nothing about the message that is being signed. The *blindness* is an important property in blind signature scheme. In this work, we analyze security of the blind signature[1], and show that the scheme hasn't blindness, in other words, the signer is able to link a valid message-signature pair obtained by some user. To overcome the above flaw, we propose an improved scheme and show that the security of the improved scheme is based on the Computational Diffie-Hellman problem.

1 Introduction

In traditional digital signature schemes, the blinding between a user and his public key needs to be ensured. The usual way to provide this assurance is by providing certificates that are signed by a trusted third party, Namely the public certificate. As a consequence, the system requires a large storage and computing time to store and verify each user's public key and the corresponding certificate. In 1984, Shamir [2]introduced the conception of identity-based public key cryptosystem to simplify key management procedures in certificate-based public key setting. In ID-based mechanism, the user's public key is indeed his identity (such as email, IP address, etc.). Since then, various ID-based encryption schemes and signature schemes have been proposed. At present, many ID-based encryption and signature schemes have been proposed based on the bilinear pairings in elliptic curves or hyper-elliptic curves. The size of signature is in general short in these schemes.

Blind signature was introduced by D.Chaum [4], which can provide an anonymity of signed message.Since it was introduced, blind signature schemes[4,5,6,7,8, 9,10] have been used in numerous application, most prominently in anonymous voting and anonymous e-cash.

Informally, blind signature allows a user to obtain signatures from an authority on any document, in such a way that the authority learns nothing about the message that is being signed. The most important property of blind signature

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 262–270, 2006.

differing from the other signatures is *blindness*, which requires that after interacting with various users, the signer S is not able to link a valid message-signature pair (m, δ) obtained by some user, with the protocol session during which δ was created. The other property is unforgeability, requires that it is impossible for any malicious user that engages in k runs of the protocol with the signer, to obtain strictly more than k valid message-signature pairs. The basic idea of most existing blind signatures is that the requester randomly chooses some random factors and embeds them to the message to be signed. The random factors are kept in secret so the signer cannot recover the message. Upon the blinded signature returned by the signer, the requester can remove the random factor to obtain a valid signature. Up to now, two ID-based blind signature schemes based on bilinear pairings have been proposed. The first scheme was proposed by Zhang and Kim[16] in Asiacrypt 2002, the other scheme[17] was proposed in ACISP2003. They claim that the security against generic parallel attack to their schemes don't depend on the difficulty of ROS-problem [18]. In fact, their scheme[17] is also forgeable under the generic parallel attack if the ROS-problem is solvable, namely, the security against generic parallel attack to this scheme also depends on the difficulty of ROS-problem.

Recently, Huang et.al proposed an ID-based blind signature scheme in CANS05 [1] (Huang et.al scheme for short) and show that the security of the scheme is based on CDH assumption (Computational Diffie-Hellman Assumpitonm), and the scheme satisfied the *blindness* of blind signature, namely, unlinkability. In this work, we show the scheme doesn't satisfy unlikability by analyzing the security of the scheme. In other words, the signer is able to link a valid message-signature pair. In this paper, we first analyze the security of Huang et.al blind signature[1], and show that the scheme hasn't blindness, in other words, the signer is able to link a valid message-signature pair obtained by some user. Finally, to overcome the above flaw, we propose an improved scheme and show that the security of the improved scheme is based on the Computational Diffie-Hellman problem.

The rest of the paper is organized as follows: Section 2 give some preliminary knowledge related to the paper; in section 3, we show the flaw of Huang et.al blind signature scheme, then propose an improved scheme in section 4; in section 5, we analyze the security of improved scheme; Finally, we draw this paper.

2 Preliminaries

In this section, we will fist review some fundamental backgrounds related to the paper.

Let G_1 be a cyclic additive group generated by P with the order prime q, and G_2 be a cyclic multiplicative group with the same order q. Let $e : G_1 \times G_1 \longrightarrow G_2$ be a pairing which satisfies the following conditions:

- Bilinearity: For any $P, Q, R \in G_1$, we have $e(P + Q, R) = e(P, R)e(Q, R)$ and $e(P, R + Q) = e(P, R)e(P, Q)$. In particular, for any $a, b \in Z_q$,

$$e(aP, bP) = e(P, P)^{ab} = e(P, abP) = e(abP, P)$$

- Non-degeneracy: There exists $P, Q \in G_1$, such that $e(P, Q) \neq 1$
- Computability: There is an efficient algorithm to compute $e(P, Q)$ for $P, Q \in G_1$.

The typical way of obtaining such pairing is by deriving them from the Weil pairing or the Tate pairing on an elliptic curve over a finite field.

Computational Diffie-Hellman Problem: Given $P, aP, bP \in G_1$ for randomly chosen $a, b \in_R Z_q$ to abP.

The success probility of any probabilistic polynomial-time algorithm $\mathcal{A}$ in solving CDH problem in G_1 is defined to be

$$Succ_{\mathcal{A}}^{CDH} = Pr[\mathcal{A}(P, aP, bP) = abP | a, b \in {Z_q}^*]$$

The CDH assumption states that for every probabilistic polynomial-time algorithm $\mathcal{A}, Succ_{\mathcal{A}}^{CDH}$ is negligible.

3 Reviews of Huang et.al Blind Scheme

In the following, we will briefly review the Huang et.al blind scheme. Please the interested reader refer to [1] for the detail content.

[Setup]
Choose a group G_1, which is a cyclic additive group generated by P with prime order q. Choose a cyclic multiplicative group G_2 with the same order q and a bilinear pairing $e : G_1 \times G_1 \longrightarrow G_2$. Pick a random $s \in_R Z_q$, set $P_{pub} = sP$. Let $H_1(\cdot), H_2(\cdot)$ be two hash functions and satisfy $H_1 : \{0,1\}^* \times G_2 \longrightarrow Z_q$ and $H_2 : \{0,1\}^* \longrightarrow G_1$. Publish the system parameter $SP = (G_1, G_2, e, q, P, P_{pub}, H_1, H_2)$, and keep the master key s secret.

[Extract]
Given an identity ID, compute $P_{ID} = H_2(ID)$ and return the corresponding private key $S_{ID=sP_{ID}}$.

[Sign]
To make the signer produce a signature, The user U first chooses $P_1 \in G_1$ and compute $e(P, P_1)$ before executing interaction. Then they execute the following interactive procedures:

1. The signer randomly chooses a number $r \in_R Z_q$, and computes
$$R' = e(P_{ID}, P_{pub})^r$$
and sends R' to the user as his commitment.
2. The user randomly chooses $t_1, t_2 \in_R Z_q$ and computes
$$R = R'^{t_1} e(P_1, P)^{t_2}$$
$$h = H_1(m, R)$$
$$h' = ht_1$$
then sends h' to the signer.

3. The signer computes
$$V' = (rh' + 1)S_{ID}$$
and sends V' to the user.
4. upon receiving V', the user checks whether the relation holds.
$$e(V', P) = R'^{h'} e(P_{ID}, P_{pub})$$
If it holds, he computes
$$V = V' + ht_2P_1$$

Then, the resultant blind signature on the message m is (R, V)

[Verify]
To verify a signature (R, V) on the message m, the verifier checks the following equation
$$e(V, P) = R^{H_1(m,R)} e(P_{ID}, P_{pub})$$

4 The Flaw of Huang et.al Blind Signature Scheme

Recently, Huang et.al presented a ID-based blind signature and claimed that their scheme satisfied the important property: blindness. Unfortunately,we show that Huang et.al blind signature scheme doesn't satisfy the blindness by analyzing the security of the scheme. Namely,after interacting with various users, the signer S is able to link a valid message-signature pair (m, δ) obtained by some user.

4.1 Linkability

Here, we will show that the signer can link a message-signature pair. According to the above signing procedure, we can know that given a blind signature (R, V) on the message m, the view of the signer is (R', h', V') in the generation of the blind signature. In the following , we will show how the signer link the message-signature pair by the views (R', h', V'). Given a blind signature (R, V), the signer executes as follows

– Firstly, the signer computes $\alpha = e(V - V')$
– then, the signer computes $\beta = R'^{h'}$
– compute $h = (m, R)$
– Finally, check whether the relation equation
$$\alpha \cdot \beta =? = R^h \tag{1}$$

if the equation (1) holds, then it denotes that the signer can link the message-signature pair.

4.2 Correctness

In the following, we will show that why a blind signature (R, V) satisfies the equation (1) above, then it means that the signer can link the message-signature pair.

Theorem 1. *Given a blind signature (R, V) on message m, the signer can link a message-signature pair by using the equation (1).*

Proof. according to the blind signature above, we know

$$V = V' + ht_2P_1,$$

thus the signer can compute

$$\begin{aligned} \alpha &= e(V - V', P) \\ &= e(ht_2P_1, P) \\ &= e(P, P_1)^{ht_2} \end{aligned}$$

and since the signer possesses R' and h' and $h' = ht_1$, then he is able to compute

$$\beta R'^{h'} = R'^{ht_1}$$

Thus, we obtain the following relation

$$\alpha \cdot \beta = R'^{ht_1} e(P, P_1)^{ht_2} = R^h$$

Note that $h = (m, R)$. According to the state above, the signer can link a message-signature pair. In other words, it indicates that the blind signature hasn't blindness. The signer can link the signature and message by his restored messages.

5 Improved Scheme

In this section, To modify the flaw of Huang et.al blind signature scheme, we give an improved blind signature scheme. The system **Setup** phase and **Extract** phase in our proposed scheme is the same ones of Huang et.al scheme. In the following, we only consider **Signing** phase and **Verification** phase.

[Signing phase]
To obtain a blind signature on the message m, a user can first chooses a point $P_1 \in G_1$ and compute $e(P_1, P)$ beforehand outside of the signing protocol. Then the signer executes the following interaction procedure with the user.

1. the signer randomly chooses a number $r \in_R Z_q$, then computes

$$R' = e(P_{ID}, P_{pub})^r$$

   and sends R' to the user as the commitment.

2. The user randomly chooses three numbers $t_1, t_2, t_3 \in Z_q$ as blinding factors, and compute as follows

$$\begin{aligned} R &= R'^{t_1} e(P_{ID}, P_{pub})^{t_2 t_1} e(P_1, P)^{t_3} \\ h &= H_1(m, R) \\ h' &= t_1^{-1} h + t_2 \end{aligned}$$

   and sends h' to the signer as the challenge.
3. The signer computes

$$V' = (r + h') S_{ID}$$

   and returns V' to the user.
4. The user computes

$$V = t_1 V' + t_3 P_1$$

   Note that $V = (h + rt_1 + t_1 t_2) S_{ID} + t_3 P_1$

Then (V, R) is the blind signature of the message m.

[Verification]
After receiving the blind signature (V, R), a verifier checks as follows:

$$h = H_1(m, R) \tag{2}$$

$$e(V, P) = R \cdot e(P_{ID}, P_{pub})^h \tag{3}$$

if the equations above (2) and (3) hold, the verifier accepts it as a valid blind signature.

6 Security Analysis of the Proposed Blind Signature Scheme

In the following, we first show that the proposed scheme satisfies correctness.

According to the signature above, given a blind signature (R, V), we know

$$\begin{aligned} h &= H_1(m, R) \\ e(V, P) &= e(t_1 V' + t_3 P_1, P) \\ &= e(t_1 V', P)(P_1, P)^{t_3} \\ &= e(t_1 (r + h') S_{ID}, P)(P_1, P)^{t_3} \\ &= e((h + rt_1 + t_1 t_2) S_{ID}, P)(P_1, P)^{t_3} \\ &= e(P_{ID}, P_{pub})^{(h + rt_1 + t_1 t_2)} (P_1, P)^{t_3} \\ &= e(P_{ID}, P_{pub})^h R \end{aligned}$$

Obviously, the blind signature (R, V) on the message m satisfies the verification equation, thus it indicates that the scheme satisfies correctness.

Let an adversary $\mathcal{A}$ be a probabilistic polynomial time Turing machine whose input only consists of public data $(G_1, G_2, e, P, P_{pub}, H_1, H_2)$. $\mathcal{A}$ can make q_s queries to the signer, and q_{H_1} queries to the random oracle H_1.

Theorem 2. *If there exists an adversary $\mathcal{A}$ can forge a blind signature in polynomial time with non-negligible probability $\epsilon \geq 10(q_s + 1)(q_s + q_{H_1})/2^l$, then the CDH problem can been solved in G_1 in polynomial time with non-negligible probability.*

Proof. (sketch). Suppose that $\mathcal{B}$ is a CDH attacker. Given an instance (q, P, aP, bP), Let $\mathcal{A}$ be a forger that breaks the proposed signature scheme under chosen message attack. We show how $\mathcal{B}$ can use $\mathcal{A}$ to the CDH problem, i.e. to compute abP.

First, the challenger $\mathcal{B}$ sets $P_{pub} = aP$ and (q, P, aP) to the forger $\mathcal{A}$ the system public key. Let bP be the hash value ID which is the identity of the signer, namely, $H_2(ID) = bP$. If $\mathcal{A}$ can forge a valid signature (m^*, R^*, V^*, h^*) with running time t in a non-negligible $\epsilon \geq 10(q_s + 1)(q_s + q_{H_1})/2^l$ under adaptively chosen message.

Apply the "forking Lemma" formalized in [15], $\mathcal{B}$ can obtain two valid blind signatures (m^*, R^*, V^*, h^*) and (m^*, R^*, V'^*, h'^*) such that $h^* \neq h'^*$. Then they satisfy the following relation.

$$e(V^*, P) = R^* \cdot e(P_{ID}, P_{pub})^{h^*} \tag{4}$$

$$e(V'^*, P) = R^* \cdot e(P_{ID}, P_{pub})^{h'^*} \tag{5}$$

Thought the above equation (3) and (4), we have

$$e(V^*, P) \cdot e(V'^*, P)^{-1} = e(P_{ID}, P_{pub})^{h^*} \cdot e(P_{ID}, P_{pub})^{-h'^*} \tag{6}$$

Thus, we can obtain

$$e(V^* - V'^*, P) = e(P_{ID}, P_{pub})^{h^*} \cdot e(P_{ID}, P_{pub})^{-h'^*} \tag{7}$$

$$e(V^* - V'^*, P) = e(P_{ID}, P_{pub})^{h^* - h'^*} \tag{8}$$

At last, the algorithm $\mathcal{B}$ can output $abP = \frac{1}{h^* - h'^*}(V^* - V'^*)$. It means that the algorithm $\mathcal{B}$ can solve an instance of Computational Diffie-Hellman problem in G_1 in excepted time. □

7 Conclusion

ID-based public key crypt-system can be an alternative for certificate-based key infrastructures. Blind signature plays an important role in secure e-commerce, such as e-cash, e-vote. Where The *blindness* is an important property of blind signature scheme. In this paper, we first analyze the security of Huang et.al blind signature[1], and show that the scheme hasn't blindness, in other words, the signer is able to link a valid message-signature pair obtained by some user. To overcome the above flaw, we propose an improved scheme and show that the security of the improved scheme is based on the Computational Diffie-Hellman problem.

Acknowledgements

The author would like to thank the anonymous referees for their valuable comments and suggestions that improve the presentation of this paper. And this work is supported by the Doctor Research starting and Scientific Research Common Program of Beijing Municipal Commission of Education (KM200610009011).

References

1. Z.J.Huang, K.F.Chen and Y.M Wang, Efficient Identity-Based Signatures and Blind Signatures, CANS2005, LNCS 3810, pp. 120-133, 2005, springer-verlag
2. A.Shamir, Identity-based cryptosystems and signature schemes, In: Advances in Cryptology-Crypto'84, LNCS 196, pp 47-53, 1985, springer-verlag
3. D.Boneh, M.Franklin, Identity-based encryption from the Weil Pairing, In: Advances in Cryptology-Crypto 2001, LNCS 2139, pp 213-229,2001, springer-verlag
4. D.Chaum, Blind signature for untraceable payment, in Advances in Cryptology-Crypto'82, 1983, pp 199-203, springer-verlag, Berlin Heidelberg.
5. Sherman S.M. Chow, Lucas C.K. Hui, S.M.Yie and K.P.Chow, Two Improved Partially Blind Signature scheme from Bilinear Pairings, ACISP2005, LNCS 3574, pp 316-328, 2005,springer-verlag, Berlin Heidelberg
6. Jinho Kim, Kwangjo Kim, and Chulsoo Lee. An Efficient and Provably Secure Threshold Blind Signature, Internal conference on Information security and Cryptology-ICICS2001, LNCS 2288,pp318-327, springer-verlag, Berlin Heidelberg
7. Torben P.Pderson, Distributed Provers with Applications to Undeniable Signatures, Advances in Cryptology-Eurocrypt'91, LNCS 547, pp 221-242, springer-verlag, Berlin Heidelberg
8. A.Boldyreva, Efficient threshold signature, multisignature and blind signature schemes based on the Gap-Diffie-Hellman group signature, PKC2003, LNCS 2139, pp 31-46 , 2003, Springer-Verlag
9. Shuhong Wang, Feng Bao, Robert H. Deng, Cryptanalysis of a Forward Secure Blind Signature Scheme with Provable Security, Information and Communications Security: 7th International Conference, ICICS 2005.
10. Jan Camenisch, Maciej Koprowski, Bodgan Warinschi, Efficient Blind Signatures Without Random Oracles,4th International Conference, SCN 2004, Amalfi, Italy, pp134-146, 2004
11. C.Dwork and M.Naor, An efficient existentially unforgeable signature scheme and its applications, Advances in Cryptology-Crypto'94, LNCS 839, 1994, pp 234-246.
12. S.Even, O.Goldreich and S.Micali. On-line/off-line digital signatures, Journal of Cryptology, Vol(9) 1996, pp 35-67
13. A.Perrig. The BiBa one-time signature and broadcast authentication. the 8th ACM Conference on Computer and Communication security, ACM, 2001, pp 28-37
14. T.Okamoto, A.Inomata and E.Okamoto, A proposal of short proxy signature using pairing, In the proceedings of the International Conference on Information Technology: Coding and Computing, pp. 631-635, 2005
15. David Pointcheval, Security Arguments for Digital Signatures and Blind Signatures,Journal of Cryptology, Volume 13, Number 3, pp361 - 396
16. Fangguo Zhang, Kwangjo Kim , ID-Based Blind Signature and Ring Signature from Pairings ,Advances in Cryptology - ASIACRYPT 2002: 8th International Conference on the Theory and Application of Cryptology and Information Security, pp533 - 547, 2002 Springer-Verlag

17. F.Zhang, K.Kim, Efficient ID-based Blind Signature and Proxy signature from Bilinear Pairings, In:Proc.of ACISP 2003, LNCS 2727, pp 312-323, 2003
18. C.Schnorr, Security of Blind discrete log signature against interactive attacks, In: Information and Communications Security-ICICS 2001, LNCS, 2299, pp 1-12
19. Xun Yi, An Identity-Based Signature Scheme From the Weil Pairing, IEEE Communications Letter, Vol.7, No2, pp 76-78, 2003

A Noble Structural Model for e-Learning Services in Ubiquitous Environment*

Minseong Ju[1], Seoksoo Kim[1,**], Yeong-Deok Kim[2], and Sukhoon Kang[3]

[1] Dept. of Multimedia Engineering, Hannam University,
133 Ojeong-Dong, Daedeok-Gu, Daejeon 306-791, Korea
nimpe2@naver.com, sskim@hannam.ac.kr

[2] Dept. of Computer Information Science & Engineering, Woosong University,
17-2, Jayang-Dong, Dong-Ku, Daejeon 300-718, Korea
ydkim@wsu.ac.kr

[3] Department of Computer Engineering, Daejeon University,
96-3 Yongun-Dong, Dong-Gu, Daejeon 300-716, Korea
shkang@dju.ac.kr

Abstract. As e-learning studying is activated, learners' requirements are increased. It is important to note that the effective e-learning model augmented requirements of learner and new ubiquitous environment are artifacts of an era of u-learning. This paper has analyzed learners' requirements and limitations in the existing e-learning system, and proposed the addition of contents conversion service and collaborative learning service to LMS based on SCORM standard proposal using ubiquitous network, next-generation sensor technology, etc. in order to construct effective and unbounded u-learning system in ubiquitous environment. Based on this structural model. we also propose XML-MCAS security method suitable for wireless environment for preventing the leakage of personal and contents information.

1 Introduction

As for communication platform, PC and the Internet have been the main components of learning environment in the past, but now in the age of ubiquitous environment we are in the need of a new e-learning model. From now, e-learning in ubiquitous environment is called u-learning for short. Ubiquitous environment will give learners learner-centered creative educational environment, and allow them to learn any contents at any time and in any place freely and conveniently. This is the goal pursued by the u-learning. In order for the current e-learning system to evolve to u-learning system, we need to modify and improve LMS (Learning Management System), which is the root of the system structure.

This paper has analyzed learners' requirements and limitations in the existing e-learning system, and proposed the addition of contents conversion service and

* "This work was supported by a grant No. (R12-2003-004-03003-0) from Ministry of Commerce, Industry and Energy"

** Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 271 – 278, 2006.

collaborative learning service to LMS based on SCORM standard proposal using ubiquitous network, next-generation sensor technology, etc. in order to construct effective and unbounded u-learning system in ubiquitous environment. Moreover, to solve the vulnerability of security in ID-password system, we introduced XML-MCAS security method suitable for wireless environment for preventing the leakage of personal and contents information.

Problem Definitions in Traditional e-Learning Approaches:

(1) Limitations in experiential (active) learning: Learning activities in the context of e-learning are largely divided into passive learning in which learners study information prepared by specialists following the logic prescribed by the specialists and active learning in which learners search various resources related to learning contents. The former is called push learning and the latter is pull learning. Existing e-learning was mostly push learning or spray learning that sprays a large volume of information [8]. These types of learning activities can create only explicit knowledge. Explicit knowledge means that the knowledge which is easy to express and explain verbally like documented knowledge and processes described in the form of manuals. E-learning occurs in a digital space. Different from an analogue space, a digital space cannot give practical experiences of applying what to feel and realize to real situations.

(2) Limitations in collaborative learning: In the current form of e-learning, teachers and learners have few chances to meet each other face-to-face. They meet via text messages through a bulletin board. Sometimes teleconference system is used but its use is highly restricted by the system performance and various limitations. Thus, it is not easy to execute group projects and collaborative learning through e-learning. Thus, still learning through classroom lectures is considered superior to e-learning. That is, e-learning system is inferior to traditional learning in that it cannot expect solidarity among learners and team meetings for creating new ideas.

(3) Inappropriateness of contents evaluation: In order for learners to study through e-learning, the quality of contents must be guaranteed. High-quality contents may be misunderstood as various multimedia functions and gorgeous appearance. People believe that learners' curiosity should be stimulated and their attention should be drawn with displays looking colorful and dynamic at a glance. In a sense, this is reasonable considering the peculiarity of online learning. However, learning effect through e-learning is maximized not by seemingly gorgeous and dynamic contents but by elaborately designed internal logical structure that induces learners to make learning activities in connection with learning strategies optimized for the nature of learning contents. That is, the quality of contents should be evaluated not by visible attractiveness but by the logical structure of the learning contents and the dynamic inducement of learning activities for substantial and durable learning effects.

(4) Lack of security: Because E-learning exchange information and contents over the network between learners and teachers, the risk of information leakage is quite high. However, existing e-learning service is fully exposed to security risk as it identifies users just with ID and password and distinguishes the access rights of learners, teachers and administrators. Thus, if a user's ID and password is obtained, one can access the system without any other authentication process and, as a result, users' personal information and various data and contents for only authorized users are left unprotected. On the other hand, considering the characteristic of e-learning that continues information transmission without interruption, high security may increase users' inconvenience. Moreover, because learning services in ubiquitous environment are mainly provided in wireless mode, authentication system must be improved for wireless environment.

2 Related Works

2.1 Ubiquitous Environment and e-Learning

Computer environment is switching from the age of PC to the ubiquitous age. Ubiquitous environment is a computer environment, in which various types of computers are embedded in everyday life and are used freely and conveniently whenever necessary like water and electricity. That is, unlike in the previous paradigm where humans provide computers sensing and interface functions, in the new paradigm, computers will sense necessary functions and provide interface customized to users. The ideal ubiquitous computer is "unbounded computer" and "autonomous computer." As networked micro-computers are implanted in things and places, people can get information from any place [1]. Next-generation technology developed by various companies and U-Korea Plans promoted by the Korean government show that the ubiquitous age is being unfolded before our eyes. As a part of U-Korea Plans, the Ministry of Information and Communication is making intensive research on four key technologies: IPv6-based low-power WPAN technology interoperable with mobile networks; UWB technology for 100Mbps low-speed sensing; high-speed multi-sensing RFID technology; and intelligent wireless sensor network for routing. In particular, RFID (wireless sensing, Radio Frequency Identification) is a technology for identifying, tracking and managing objects, animals or humans carrying a micro-chip containing identification information using wireless frequency, and is being applied in various areas including logistics, distribution, electronic payment and security [2].

2.2 Sensing Technology in Ubiquitous Environment

In ubiquitous environment, physical and chemical conditions of the human body and its environment such as light, temperature, smell and weight are recognized by using various sensors. Data collected by a sensor is processed by a micro-processor. These technologies have already been developed and commercialized. Sensed signals are converted to electric signals by a signal controller. Data converted to electric signals are digitized by an A/D converter and put into the microprocessor and then the data is transformed to information through the embedded operating system.

Sensors are largely divided into two categories. One is manual sensing system in which a reader senses an identification chip planted in an object. The system performs the sensing function according to a standard method agreed upon between the identifier and the reader. A standard interface is applicable between the subject and the object. Representative RFIDs are Active Badge and 2D Barcode. The other is sensing system in which sensors, like the five human senses, recognize the environmental changes and send information by them.

Table 1. Sensing systems corresponding to the five senses

Sense	Object	Man	Sensor	Interface
Vision	Visible rays	Eye	Image sensor	Motion recognition
Hearing	Sound	Ear	Sound sensor	Voice recognition
Touch	Mechanical energy	Skin	Touch sensor	Touch screen
Smell	Chemical elements	Nose	Gas, bio sensor	Under research
Taste	Chemical elements	Mouth	Ion, bio sensor	Under research

As in Table 1, sensing systems have outstanding sensing abilities through operation similar to human sense organs but current technology is not enough to sense smells and food tastes [3]. Intelligent interface is important and non-standard approaches are required. A representative technology is Small Dust. This technology scatters dust-size sensors around the physical spaces like buildings, roads, clothes and bodies and collects information on temperature, humidity, acceleration, pressure, etc. through wireless network. Inside a smart dust are sensor, sensor control circuit, computer, bidirectional communication module, power supply, etc. With the advance of contemporary VLSI semiconductor technology and MEMS (Micro Electro Mechanical System) technology, it is possible to implement such a device as small as a grain of sand [4, 5].

3 A Refined e-Learning Model in Ubiquitous Environment

3.1 SCORM-Based u-Learning Model

From computer-based learning (CBI) to Web-based learning (WBI), the development of classes in computer environment demands huge investments of time and money. In order to overcome the inefficiency in development, e-learning researchers are looking for the reuse of developed contents and the sharing of contents developed by third parities. By developing a system for reusing a part or the whole of existing contents or sharing contents created by third parties, we can save a lot of time and money. Such efforts have been integrated into the establishment of e-learning technology standard. SCORM proposed by ADL in the U.S. is evolving into a form integrating general standards [6]. Many e-learning-related products are being introduced in the market including e-learning platforms, authoring tools and contents development tools. E-learning technology standard set definite guidelines for contents and platforms so that contents can be reused and shared independently from the platform and platforms can easily interoperate with one another. Contents are created as learning objects containing texts, graphics, videos, sound files, etc. and the objects are stored, searched and delivered. According to e-learning relevant literature, learning objects are also described as reusable learning objects, sharable contents, etc. but 'learning object' is the most common term. ADL SCORM is generally accepted as technology standard by many relevant companies.

In order to implement requirements of SCORM satisfactorily, we must have Web-based LMS that executes contents developed by different companies and searches contents in database. LMS is composed of functions designed to manage learning contents, execute learning and track learners' responses. LMS can be applied to simple class management as well as extremely complicated wide-area distributed environment. SCORM defines interoperation between contents and LMS environment and does not specify functions implementing specific LMS.

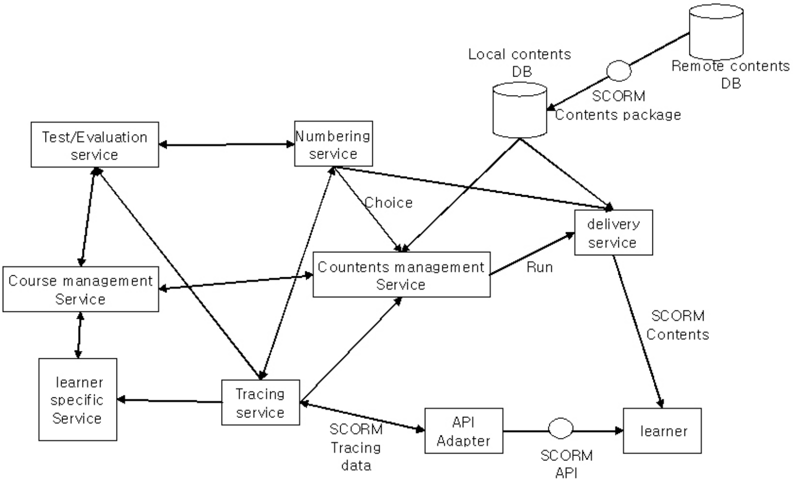


Fig. 1. Structure of SCORM-based LMS

Figure 1 shows the traditional components and services of SCORM-based LMS. LMS is a method to delivery learning contents to learners and contains a number of services such as deciding what and when to deliver (delivery), tracking learning process through learning contents (tracking) and setting the order of delivery to learners according to predefined rules (sequencing) [7]. Different from previous CBI system, 'learner-specific service' and 'tracking service' provide information for establishing adaptable learning environment. LMS collects information on learners' characteristics, delivers contents, monitor learners' responses and achievements through contents, and helps learners decide what to study next.

In order to overcome limitations in e-learning analyzed in Example 1 and prepare the coming ubiquitous age, we propose a number of services to be added to current SCORM standard, based on LMS, and the IEEE standard learning management system.

3.2 Learning Form Conversion Service

Learning form conversion service senses a learner's surrounding situations and recognize his/her body condition using smart sensor technology and provides the learner with contents in the optimal form. LMS has four characteristics. First, the teaching designer creates a new learning object or a new course by combining existing learning objects. Second, the editor reviews the learning object or course submitted by the designer and approves it. Third, individualized rules adjusted to the learner. Lastly, old learning objects and courses are stored in the archives or deleted. The present study proposes to add learning form conversion service and collaborative learning service. The reason for the proposal is that services necessary for u-learning can be added by the characteristic “the editor reviews a submitted learning object and approves it” mentioned above. Because learning contents are designed and implemented at the unit of learning object, it can be created promptly and its reuse and reshuffling can be fast and efficient. Learning form conversion service recognizes the learner’s current condition using smart sensors, searches database to find services in the form optimal for the learner, links the learner to the learning contents provider or to a learning service server of appropriate form.

3.3 Collaborative Learning Service

The biggest weak point of current e-learning is its limitation in collaborative learning. In ubiquitous environment, however, we can implement U-learning that solves the problem. We made this research based on the technology of Orestia project, SOB project and Paper++ project. When these projects are completed, the systems can be constructed in connect with the present research. Orestia project is to design, develop and evaluate modules and symbol - sub-symbol structures of intelligent artificial objects mainly focused on interaction with humans. If the Orestia architecture is put in specific environment requiring certain features, it retrains the sub-symbol neural network of an artificial object so that the object is bestowed with new features. Combing existing technology with new wireless communication, it provides a common protocol and format that enable exchange of different types of data among artificial objects and between artificial objects and service providers [9]. That is, using Orestia, modules and multi-purpose information objects are created for interaction among learners in the same group. SOB project aims to develop effective sound and sensory models based on physical acoustic phenomena happening in artificial objects and equipment having physical interaction with humans. Variables in the sound model are human body motions and gestures, which are managed by a control model. That is, although it is a learners’ meeting in virtual space, they feel handshake by touching the video screen and sense the texture of real objects. Paper++ project develops stored learning materials embedded with sensors and position-based devices, aiming to improve the useful properties of paper using sensors, in order to fill the gap between materials and the electronic domain. If the participants finish their discussion, the minutes of the meeting are automatically saved and delivered to all the learners of the group [9, 10]. Using ubiquitous projects like Orestia and SOB, users can shake hands as if they are meeting face-to-face and feel objects as if they are real things. Moreover, using Paper++, members can receive the contents of discussion that are automatically saved and delivered.

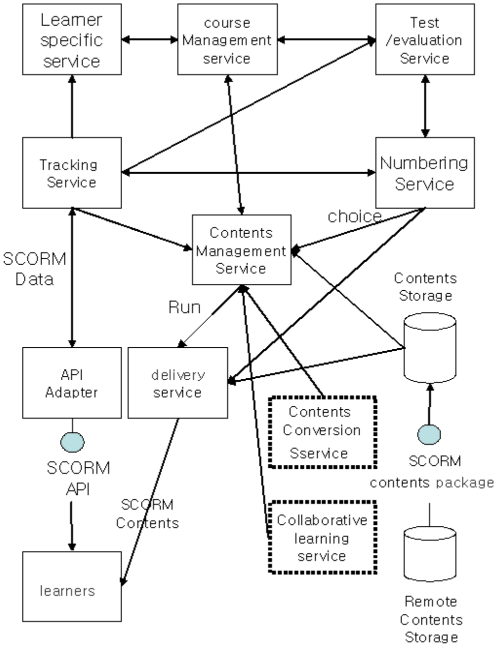


Fig. 2. LMS structure proposed for u-learning

The reasons for proposing the two services to be added to SCORM-based LMS can be explained in the following five aspects with regard to the advantages of SCORM standardization as follows: First, contents are interoperable with other different systems without additional work. Second, objects can be reused in various ways. Third, the system can track and manage learners and contents, collecting necessary information. Fourth, learners can get information on contents and access them at convenient time and access from any place. Fifth, the standard is durable to new technologies and products.

4 Security of Distance Education Model in U-Learning

The U-learning model proposed above was embedded with various sensing systems for its application at new ubiquitous environment, and its insufficiencies were supplemented in current e-learning system. However, such a system must consider security. Research on problems in existing e-learning has been focused on the structure of teaching and learning system. In distance education system that transmits various types of data and information between the teacher and the learner, however, personal and contents information is exposed to high risk of leakage. Nevertheless, many e-learning systems just use the ID and password system on the Web. If one acquires a user' ID and password, information on teachers, learners and contents involved in the e-learning systemcan be easily disclosed.

To solve such a security problem, we need to provide a more advanced authentication service to teachers and learners using the system, but the commonly used PKI-type authentication system is not suitable in ubiquitous environment containing many mobile systems and in U-learning demanding real-time connection and smooth information transmission. Thus, it is considered effective to prevent information leakage by building MCAS (Multiplex Certification Agent System) that adopts XML electronic signature.

5 Conclusions

The wave of the ubiquitous age is now rising from the feet. In addition, our social conditions accelerate the increase of demands from e-learning learners and, as a consequence, limitations in e-learning as well as learners' new requirements are being found. Facing the new age, therefore, we propose a new model for implementing U-learning improved by reflecting e-learning learners' requirements. The three characteristics of the proposed u-learning model are as follows. First, it adopts sensing technologies in ubiquitous environment, which are under research. Second, it provides learning form conversion service and real-time collaborative learning service. Lastly, it gives interoperability, reusability, controllability, accessibility and durability without extra work as it is added to SCORM-based LMS. What is more, in order to reinforce the security of existing e-learning solutions, we suggested information security using XML-MCAS, which is fit for wireless systems in ubiquitous environment. Sensing systems and security technologies in ubiquitous environment need to be studied and improved further in the future. The outcomes of this study may be usable depending on the progress of ubiquitous technologies.

References

1. Mark Weise, "The Computer for the Twenty-First Century," Scientific American, pp. 94-101, September 1991.
2. "Ubiquitous," Korea Information Processing Society, Vol. 10, No. 2, June 2003.
3. "Sensor-Tech/Report," Electronics and Telecommunications Research Institute, 2003.
4. Bung-Ki Son, "Sensor Engineering," Iljin 2002.
5. Woo-Hyuk Park, "FI_Report," Hyun-Dai person development cyber education center, 2003.
6. Ho-won Jung, Bum-jin Lee, "SCORM Practices Guide for Content Developers", 2004.
7. In-Ho Choi, "LMS Construction Guide Manual for Cyber Studying System Support", KERIS Report, 2004.
8. "e-Learning White Paper," Chapter 4 KAOCE, http://www.kaoce.org, August 2004.
9. Ian F. Akyildiz et al,"A Survey on Sensor Networks," IEEE Communication Magazine, August 2002.
10. Jung-Kook Lee, "World Ubiquitous Computing Strategy," October 2003.

Backward Channel Protection Method for RFID Security Schemes Based on Tree-Walking Algorithms

Wonjoon Choi[1] and Byeong-hee Roh[2]

[1]Digital Media Car AV S/W Group, LG Electronics Inc
mecgebi@lge.com
[2]Graduate School of Information and Communication, Ajou University,
San 5 Wonchon-dong, Youngtong-gu, Suwon, 443-749, Korea
bhroh@ajou.ac.kr

Abstract. Most RFID (Radio Fequency IDentification) Tag security schemes assumed that the backward channel from tags to a reader is safe from eavesdropping. However, eavesdroppers near a tag can overhear message from the tag illegally. This may cause some privacy issues because the backward channel eavesdropping means the expose of personal information related to the tags that each person has. In this paper, we propose a method to protect the backward channel from eavesdropping by illegal readers. The proposed scheme can overcome the problems of conventional schemes based on tree-walking algorithm. It is shown that the proposed method can provide the probability of eavesdropping in some standardized RFID tag system such as EPCglobal, ISO, uCode near to '0'.

1 Introduction

Most of currently used RFID tag type is passive. Passive RFID tag does not have its own power, and the power is supplied by external devices such as readers. Accordingly, passive tag can not initiate communication with a reader, but it can send the information encoded in its memory, e.g. tag ID, only after the reader sends power and query signals to it.

In RFID applications, a reader has to distinguish individual tag IDs in its reading region. However, since several tags in the region may respond simultaneously to the reader's query, signals from tags are interfered with each other, and then the reader can not recognize any tag ID. This situation is called collision. Several anti-collision schemes to avoid such a collision situation have been proposed. Tree-walking algorithm is one of those anti-collision schemes[1]. In tree-walking algorithm, however, since a reader sends its bit-by-bit query signals for distinguishing each tag ID, any eavesdropper within the reader's signal transfer range can also detect tag IDs by monitoring the query sequence from the reader. This may cause severe privacy problems.

To solve the eavesdropping problem in tree-walking algorithm, some schemes such as silent tree-walking and randomized tree-walking[2] algorithms have been

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 279–287, 2006.

proposed. These schemes can protect signals on the forward channel from a reader to tags only. However, if an eavesdropper is located very near to a tag, it can overhear the short range signals through the backward channel from tag to reader, and obtain the entire ID of the tag. This may cause some privacy issues because the backward channel eavesdropping means the expose of personal information related to the tags that each person has.

In this paper, we propose a simple but effective method to solve the backward channel eavesdropping problem in randomized tree-walking algorithm for RFID security. The proposed method can be applied to all kind of schemes that tags provide their ID information to readers as in the randomized tree-walking algorithm. The proposed method can make the eavesdropping probability on the backward channel near to '0' for those standardized RFID systems such as EPCglobal[3], ISO/IEC [4], and uCode[5].

The rest of the paper is organized as follows. In Section 2, some works related to the proposed method and their problems is briefly reviewed. Then, we will explain our proposed scheme in Section 3, and its performance will be illustrated in Section 4. Finally, we will conclude the paper in Section 5.

2 Related Works

2.1 Binary Tree-Walking Algorithm

Overview of Tree-Walking Algorithm. In binary tree-walking algorithm[1], each tag has two states: on and off. Tags in on state can respond to a query from a reader, while tags in off state can not. At the beginning of query, after the reader sets all tags to be on states, it queries all tags about their IDs. Since all tags are on state at the time of the initial query, all tags respond to the query. Then, the reader checks the first bit of its received signal whether collision occurs. If there is a collision in the first bit, it means there are tags with different values in their 1st bit position. When collision occurred, the reader sets all tags with 0 (or 1) in their first bit to be off state. Then, the reader queries again. Because only tags with 1 (or 0) in their first bit position, there is no collision in the first bit. So, the reader checks the collision in the second bit. The same procedures are done repeatedly until it reaches to the last bit position. When it moves down to the last bit position, there remains only one tag in on state, and the reader can recognize the last remaining tag's ID. Likewise, as in normal binary tree construction procedure, a binary tree with each tag's ID as leaf node can be made, and then all tags' IDs can be distinguished by the reader.

Eavesdropping in Tree-Walking Algorithm. Let define the *forward channel* as the channel for signals from a reader to tags, and the *backward channel* as the channel from a tag to the reader. Using the forward channel, the reader can supply powers and send query signals to tags, while tags respond to the queries via the backward channel. Since readers supply powers to tags via the forward channel, the strength of signals through the forward channel should be

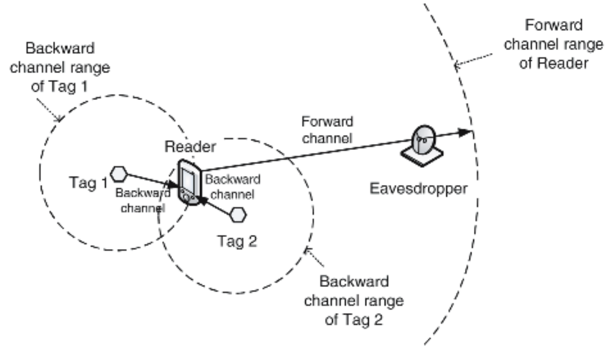

Backward channel range of Tag 1
Forward channel range of Reader
Reader
Forward channel
Tag 1
Backward channel
Backward channel
Eavesdropper
Tag 2
Backward channel range of Tag 2

Fig. 1. Signal delivery ranges from reader and tags

much stronger than that of the backward channel. As a result, the signal coverage range of the forward channel is much broader than that of the backward channel as shown in Fig.1. In binary tree-walking algorithm, since a reader sends its bit-by-bit query signals through the forward channel, if any eavesdropper within the forward channel cover range monitors the query sequence, then it can also detect all tags' IDs.

2.2 Silent Tree-Walking Algorithm

In binary tree-walking algorithm, since readers send control signals to set tags off or on for every ID bits before query, eavesdropper can know tags' IDs by monitoring the readers' signals through the forward channel. To avoid this problem, in silent tree walking algorithm[2], when there is no collision in a certain bit position, instead of sending the control signal, readers send query signal for the next ID bit directly. Since long-range eavesdroppers can not hear the backward channel, they can not know the uncollided bit value. In addition, since readers send control signals using XOR with 0 and the previously uncollided bit value, the bit value obtained by eavesdroppers is different from the original one.

2.3 Randomized Tree-Walking Algorithm

In randomized tree-walking algorithm[2], each tag has two IDs: One is a real tag ID, and the other is a random ID allocated by manufacturers or generated by the tag itself. General procedure of the randomized tree-walking algorithm is as same as in binary tree-walking algorithm except for using the random ID. That is, readers send control and query signals according to the procedure defined in binary tree-walking algorithm, but tags respond based on their random ID, not on their real tag ID. Only after each tag becomes singularized, it reports its real ID to the reader through the backward channel. Because eavesdroppers far off from tags can hear only forward channel, they can gather only random IDs, not tags' real IDs. Likewise, real IDs of tags can be secured.

3 Proposed Backward Channel Protection Method

The methods described in previous Section can protect effectively against long-range eavesdropping. However, if eavesdroppers are within the backward channel range of a tag, they can obtain the entire ID of the tag even though silent or randomized tree-walking algorithms are applied. In this Section, we explain our proposed backward channel protection method for RFID security. In our proposed method, we consider the singularization phase that only one tag sends its real tag ID in the randomized tree-walking algorithm. As we mentioned before, though the singularization has been done by using a random ID, any eavesdropper within the backward channel range can hear the tag's real ID. Our proposed method makes it impossible to overhear the tag's real ID transferred through the backward channel by eavesdroppers.

We assume the following two things. First, there is no bit error on both forward and backward channels. Second, there are no tags with same random IDs within a reader's forward channel range. We can get the validity of the second assumption from the following fact. For m tags with IDs of n-bit length, the probability that more than two tags have same random ID is $\Sigma_{k=2}^{m}\binom{m}{k}\frac{1}{2^{n(k-1)}}$. Since most of standardized ID lengths are more than 64, the probability becomes zero for those ID systems.

The operation of the proposed method is as following. At the singularization phase of randomized tree-walking algorithm, since only one tag has 'on' state, there is no collision when the tag sends its real ID. At this phase, any eavesdropper within the backward channel range can hear the tag's real ID. To avoid this situation, in the proposed scheme, when the tag sends its real ID, the reader sends a randomly generated pseudo-ID simultaneously. For this, we assume the bit-timing between the tag's real ID and the reader's pseudo-ID is synchronized as in [2]. Then, collisions occur at the positions with different bit values between the two IDs. Accordingly, eavesdroppers receive an ID with some collided bits, and we will show the recovery probability from the collided ID by eavesdroppers becomes near to zero in Section 4.

An example operation of the proposed method is illustrated in Fig.2(a). It assumes ID with 8 bits length, a real tag ID of '00001111', and a pseudo-ID of '01100111'. When both the real and the pseudo IDs are transmitted at the same time, the signal delivered to both the reader and eavesdroppers through backward channel becomes '0XX0X111' where 'X' denotes the collided bit value.

It is noted that collision occurs when different bits are sent from both reader and tag simultaneously. That is, a collision when the reader sent a bit '0' means that the tag transmitted a bit '1', vice versa. Accordingly, the reader can recover the tag's real ID exactly from the collided ID information by simply replacing the collided bits with the complements of the corresponding bits in pseudo ID generated by the reader.

Fig.2(b) shows an example of the recovery process by a reader from the received collided ID of Fig.2(a). By replacing the collided bit value with the complement of its corresponding pseudo ID, the real tag ID can be exactly recovered.

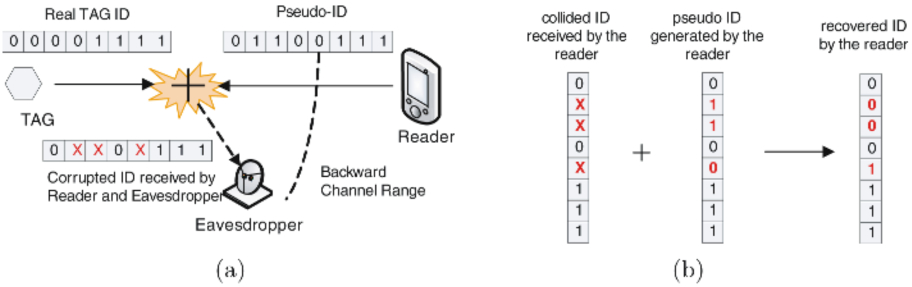


Fig. 2. Example operation of proposed method (a) collision (b) recovery

On the contrary, since eavesdroppers do not know the pseudo-ID generated by the reader, they can not recover the real ID in a same way as the reader does.

In order to implement the proposed scheme, the same communication module as in tags may be required for readers because the reader to tag modulation and encoding schemes may be different from those for the tag to reader communication as in [6]. It is noted that though each tag ID is encoded and the encoded data is delivered to the reader instead of original tag ID as in [6], the proposed scheme works well. Since encoded bit data is transmitted using a modulation scheme such as ASK or PSK, if the reader generates random bit data with the same size of the encoded data and transmits using same encoding and modulation technique used in the tag to reader communication, then collision occurs. And, the reader can recover the collided encoded data using the proposed method. Then, the reader can get the tag ID by decoding the recovered data.

4 Performance Results

In this Section, we will derive performance models for the proposed scheme, and show its degree of ID protection.

Performances of the Proposed Method. Let l be the length of a tag's ID, and p_c be the probability that a bit is corrupted by the pseudo ID randomly generated by the reader. Then, the probability that arbitrary k bits of the tag's ID with length l bits are collided with the pseudo ID, $P_c(l, k)$, can be obtained as

$$P_c(l, k) = \binom{l}{k} p_c^k (1 - p_c)^{l-k} \tag{1}$$

Let $P_f(k)$ be the probability that the corrupted ID with k collided bits is exactly recovered by the eavesdropper. And, if we let $E(l)$ be the expectation for the eavesdropper to interpret the tag's ID with l bits exactly, it is written as

$$E(l) = \Sigma_{k=0}^{l} P_c(l, k) \cdot P_f(k) \tag{2}$$

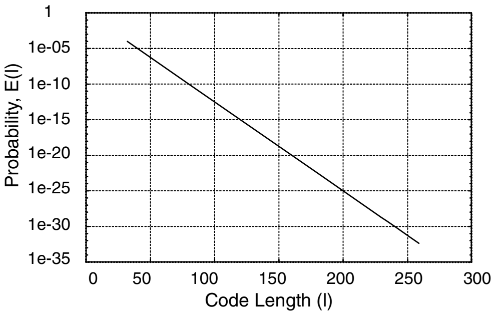


Fig. 3. Successful eavesdropping probability

Since $P_f(k) = \frac{1}{2^k}$ and $p_c = \frac{1}{2}$ in binary system, we can rewrite Eq. (2) as

$$E(l) = 2^{-l} \Sigma_{k=0}^{l} \binom{l}{k} \cdot 2^{-k} \tag{3}$$

Note that the right-hand side of Eq. (3) can be represented as a polynomial with two arguments 1 and 1/2. Accordingly, we have

$$E(l) = 2^{-l} \Sigma_{k=0}^{l} \binom{l}{k} \cdot 1^{l-k} \cdot (2^{-1})^k = 2^{-l} \cdot (1 + 2^{-1})^l = \left(\frac{3}{4}\right)^l \tag{4}$$

The intuitive meaning of Eq. (4) is as follows. Let E_c be the event that a bit of the tag's ID is collided with that of the pseudo ID, and E_t be the event that the bit value interpreted by the eavesdropper is correct whether the bit is corrupted or not. Then, the probability of the event E_t, $Pr\{E_t\}$, becomes $Pr\{E_t|E_c\} \times Pr\{E_c\} + Pr\{E_t|E_c^C\} \times Pr\{E_c^C\}$ according to the total probability theorem, where E_c^C means the complementary set of E_c. For the binary system, since and $Pr\{E_t|E_c\} = Pr\{E_c\} = Pr\{E_c^C\} = \frac{1}{2}$, we have $Pr\{E_t\} = \frac{3}{4}$. Because the interpretation of each bit value can be done independently of other bits, the probability that the tag ID with l bits can be correctly interpreted by the eavesdropper becomes $\left(\frac{3}{4}\right)^l$ as same as in Eq. (4).

Fig.3 shows the probability that the eavesdropper interprets the tag's ID with l bits from the corrupted ID exactly when the proposed scheme is applied. We can see the probability for the success of the eavesdropping decreases exponentially as the ID length l increases. Especially, for tag's ID lengths of 64, 96 and 128 that are suggested by the standardized RFID tag system such as EPCglobal[3], ISO/IEC[4] and uCode[5], the probabilities are 1.01×10^{-8}, 1.01×10^{-12}, 1.03×10^{-16}, respectively, which are near to '0'.

Performance Comparisons with Other Schemes. In this subsection, we compare the performances between the proposed scheme and other tree-walking based schemes such as binary tree-walking (BTW), silent tree-walking (STW), and randomized tree-walking (RTW). Let E_{FW} and E_{BW} be the events that eavesdroppers overhear the information through forward and backward channels, respectively.

Table 1. Comparisons of successful eavesdropping probabilities

Scheme	Tag ID Eavesdropping Probability
BTW	$E(l)_{BTW} = Pr\{E_{BW}\} \cdot 1 + Pr\{E_{FW} \cap E_{BW}^{C}\} \cdot 1$ (5)
STW	$E(l)_{STW} = Pr\{E_{BW}\} \cdot 1 + Pr\{E_{FW} \cap E_{BW}^{C}\} \cdot P_{STW}$ (6) where P_{STW} is ID exposure probability on forward channel (see Appendix)
RTW	$E(l)_{RTW} = Pr\{E_{BW}\} \cdot 1 + Pr\{E_{FW} \cap E_{BW}^{C}\} \cdot 0 = Pr\{E_{BW}\}$ (7)
Proposed	$E(l)_{PRO} = Pr\{E_{BW}\} \cdot \left(\frac{3}{4}\right)^{l} + Pr\{E_{FW} \cap E_{BW}^{C}\} \cdot 0 = Pr\{E_{BW}\} \cdot \left(\frac{3}{4}\right)^{l}$ (8)

In Table 1, the successful eavesdropping probabilities are shown for the four schemes. To obtain the probabilities shown in Table 1, it is considered the following two cases: One is that the eavesdropper is located in backward channel range, and the other case is that the eavesdropper is out of backward channel range but within forward channel range. For the three comparable schemes, when eavesdroppers are in backward channel range, they can get the corresponding tag ID. However, for the proposed scheme, since there are collided bits in the ID received by eavesdroppers, the probability significantly decreases to the amount of $\left(\frac{3}{4}\right)^{l}$ according to the ID length l.

When the eavesdropper is out of backward channel range but within forward channel range, binary tree-walking scheme exposes all tag IDs to the eavesdropper. In silent tree-walking algorithm, when no collision occurs at a bit query, instead of sending the control signal, readers send query signal for the next ID bit directly. This makes that eavesdroppers can not know the exact ID bit values of those uncollided bits. Thus, the ID eavesdropping probability for this case decreases as shown in Eq.(6). The derivation of the probability for silent tree-walking scheme is shown in Appendix. For the randomized tree-walking and the proposed schemes, since meaningless bit information is delivered through the forward channel, the ID eavesdropping probability approaches to 0.

From Table 1, the tag ID eavesdropping probabilities for those schemes have the following relationship

$$E(l)_{BTW} \geq E(l)_{STW} > E(l)_{RTW} > E(l)_{PRO} \tag{9}$$

It is noted that the ID eavesdropping probabilities such as $E(l)_{BTW}$, $E(l)_{STW}$ and $E(l)_{RTW}$ are independent of ID length, while $E(l)_{PRO}$ decreases exponentially as ID length increases.

Summary on Basic Characteristics of Comparable Schemes. In Table 2, basic characteristics of the schemes compared in Table 1 are summarized. The proposed scheme requires additional overhead in readers such as the generation of random ID and the recovery of collided bits, but it can provide high security for the backward channel that is not provided by other schemes. It is noted that the eavesdropping on backward channel means the expose of personal information related to the tags that each person has.

Table 2. Basic characteristics of compared schemes

feature	forward channel protection	backward channel protection	additional reader overhead	additional tag overhead
BTW	no	no	none	none
STW	partial protection depending on ID distribution	no	none	none
RTW	meaningless information on forward channel	no	none	(random ID)+(actual ID transmission at singularization stage)
Proposed	same as RTW	high security	(random ID collision function)+(recovery of collided bits)	same as RTW

5 Conclusion

In this paper, we proposed a simple but efficient method to protect the backward channel for securing tag ID information and privacy in RFID environments. We derived the performance model of the proposed scheme and showed that the eavesdropping probability becomes near to '0' for those standardized RFID systems such as EPCglobal, ISO, and uCode. The performance of the proposed scheme has been also compared with other tree-walking-based schemes.

While the existing ID security schemes based on the binary tree-walking algorithm have been focused on the forward channel protection, the proposed scheme can provide the backward channel security. So, by combining the proposed method with conventional schemes, we can achieve the security on both channels. It will contribute to activate the provision of RFID-based applications.

Acknowledgements. This work was supported by grant (No. 05A3-I3-10) from Ubiquitous Autonomic Computing and Network Project sponsored by the Ministry of Information and Communication, Korea.

References

1. Jacomet M., Ehrsam A., Gehrig U.: Contactless Identification Device With Anticollision Algorithm, IEEE CSCC'99, Athens, Greece, July (1999)
2. Weis S.A.: Security and Privacy in Radio-Frequency Identification Devices, Masters Thesis. MIT. May (2003)
3. EPC global: EPCTM Tag Data Standards Version 1.1 Rev.1.24, Standard Specification, April (2004)
4. ISO/IEC 15459: Information technology - Unique identifiers for item management, (1999)
5. Ken S.: Ubiquitous ID Center has authorized 2 types of RFID chips mady by Fujitsu as the Standard ucode tag, uID Center, Dec. (2004)
6. EPCglobal: EPCTM Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz-960 MHz, Version 1.0.9, Jan. (2005)

Appendix: Derivation of Eq.(6)

Let K_n be the maximum number of possible tags that can respond to n-th bit query in silent tree-walking algorithm. Then, we have $K_n = 2^{l-n+1}$. According to the query mechanism in silent tree-walking algorithm, if the actual number of remaining tags is larger than $K_n/2$, it can not avoid the collision at n-th bit query. On the contrary, collision does not occur only if the actual number of remaining tags is less than or equal to $K_n/2$ and the n-th bits of those tags are same value. Let $P_{c,n}$ be the probability that collision does not occur at n-th bit query. If we assume the uniform distribution for the tag IDs, then we have

$$P_{c,n} = \sum_{k=1}^{2^{l-n}} \left(\frac{1}{2}\right)^k \cdot \left(\frac{k}{K_n}\right) \qquad , n = 1, 2, ..., l \tag{A1}$$

Let define $\mathbf{C} = (c_1, c_2, ..., c_l)$ with the following elements

$$c_k = \begin{cases} 1 & \text{if a collision is not occured at } n\text{-th bit query} \\ 0 & \text{otherwise} \end{cases} \tag{A2}$$

Let define S_m as the set with all possible cases that m collisions occur until one tag is selected at the singularization phase. That is,

$$S_m = \left\{ \mathbf{C} \middle| \Sigma_{k=1}^{l} c_k = m \right\} \qquad , m = 0, 1, 2, ..., l \tag{A3}$$

And, we also define $P_{S,m} \equiv Pr\{S_m\}$. Some examples of $P_{S,m}$ are as following
$P_{S,0} = P_{c,1}P_{c,2}...P_{c,l}$
$P_{S,1} = (1-P_{c,1})P_{c,2}...P_{c,l} + P_{c,1}(1-P_{c,2})...P_{c,l} + ... + P_{c,1}P_{c,2}...P_{c,l-1}(1-P_{c,l})$
$P_{S,2} = (1-P_{c,1})(1-P_{c,2})P_{c,3}...P_{c,l} + ... + P_{c,1}P_{c,2}...(1-P_{c,l-1})(1-P_{c,l})$
...

In silent tree-walking scheme, since eavesdroppers have to infer the actual values from the uncollided bits, we have the following ID exposure probability

$$P_S TW = \sum_{m=0}^{l} P_{S,m} \cdot \left(\frac{1}{2}\right)^m \tag{A4}$$

Design of the Configurable Clothes Using Mobile Actuator-Sensor Network

Bo-Hee Lee[1], Kyu-Tae Seo[1], Jung-Shik Kong[2], and Jin-Geol Kim[3]

[1] School of Electrical Eng., Semyung University, ShinWal-Dong, Chechon, Korea
{bhlee420,jjabari}@nate.com
[2] Dept. of Automation Eng., Inha University, YongHyun-Dong, Nam-Gu, Inchon, Korea
selkirk@paran.com
[3] School of Electrical Eng., Inha University, YongHyun-Dong, Nam-Gu, Inchon, Korea
john@inha.ac.kr

Abstract. This paper presents the design of reconfigurable clothes that can be shown on a fashion show with actuator and sensor network. These days, some kinds of clothes are often required to perform the multiple images by transforming the shapes of clothes. In this case, reconfigurable clothes − that is, clothes that can be reconfigured by an electronic device − can be very useful way. In this study, an embedded controller using wireless sensor network (WSN) is proposed to change the shape of the clothes and also collect the information on the show-stage and clothes configuration. To perform reconfigurable clothes, remote operator based on WSN is mounted on a jacket or a trouser and control clothes. The structure of the controller, mounting method, networking method, and the configuration method are discussed in detail. To verify our design, a fashion-show example is provided. By real performance with fashion model wearing these reconfigurable clothes, the usefulness of this method and validness of a WSN application to reconfigurable clothes is verified.

1 Introduction

Clothes are one of the most essential parts of life. They have rapidly changed and remarkably improved with the times. Moreover, clothes are changing with the industrial advancement in closed relationship with industrial infrastructure such as mass media and Internet advertisement. Clothes often express social position and superior personal characteristics, and also represent the general social system of a certain period [1]. Therefore, clothes should provide not only simple comfort or protection, but also a new image [2]; this concept is so called the new fashion generation. Clothes in the past have just shown simple design under the restriction of cloth material and source [3]. However, more specific clothes systems are required in these days. Reconfigurable clothes are a good way to express beauty of clothes, but a simple wireless system has a limitation because of the distance from base station to target the stage during fashion show. Here, a sensor network is a good solution to attack this problem. In this study, the clothes transformation system is proposed. It is used for generating design deformation and supporting adjustable clothes. To express the variety in clothes, we propose an electronic system to support configurable clothes. This system

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 288 – 295, 2006.

can be used in real life, as well as at exhibitions such as fashion shows. It contains a small operating system, WSN, and an electrical circuit related with the clothes interface. It is mainly based on TinyOS[4] from Berkeley, and some application program and hardware are included. Applications in various fields of research are being developed. Interesting ongoing projects is applied to extensive experimentation of structural response to earthquakes [5], habitat monitoring [6], and intelligent transportation systems [7]. Other important fields of applications include home and building automation, and military applications. Self-configurable, ubiquitous, easy to deploy, secure, and undetectable sensor networks are an ideal technology to employ in intelligence Operations. Some similar researches have been in the field of robot application such as CotsBot[8], Robomote[9], MAS-mote[10] and tiny mobile robots. They use a WSN as a controller and TinyOS as an operating system and a hardware body as a robot. This paper suggests a clothes expression method by using WSN to reveal the beauty of fashion clothes and flexibility of designs on a fashion show stage.

2 Overall Configuration

The configurable clothes can be applied to express the multiplicity and beauty of clothes in the fashion-show stage. Many fashion models are wearing the various clothes and producing various expressions on the stage. In this case, transformation of their clothes is a good way to show the diversity. Figure 1 depicts the fashion-show stage example with configurable clothes. The fashion models are wearing the clothes with the configurable mechanism including WSN, which are exchanging their information related with adjustable configuration times and configuration extents. In case of using the traditional method with wire communication or non-electrical method, there is no method to share the stage information because they have to continuously move around. So it is required to have WSN. In this paper, some reconfigurable clothes with WSN is suggested, which is applied in fashion-show stage. As seen in the figure, node #2 and node #3 are exchanging their information in near area, but nodes from #4 to #6 do not communicate with each other since they exist in rather long distance. Of course this situation is a basic form of ad-hoc networking and varied with time.

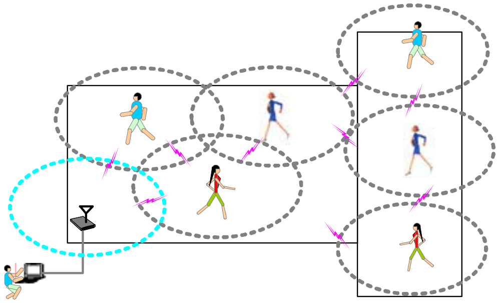


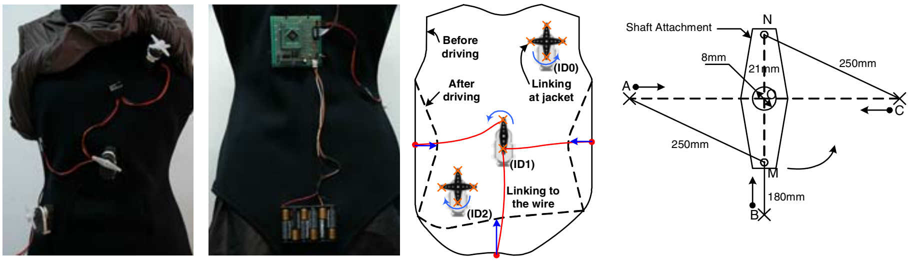
Fig. 1. Conceptual fashion-show stage with WSN

The fashion models wearing the clothes change their position during fashion-show period continually. Finally node #1 is connected with a remote operator, who controls the fashion-show performance via its base node in the remote area.

3 System Design

3.1 Clothes Design

A jacket and a trouser as clothes samples are designed to test the concepts. Figure 2 depicts a jacket mounting shape of the attached system. As shown in figure, external shape of cloth is focused on a beauty rather than a function. However the cloth material is required to make with span material to withstand deformation of the cloth and maintain convenience, and also the cloth should be strong to mount the controller and some battery.

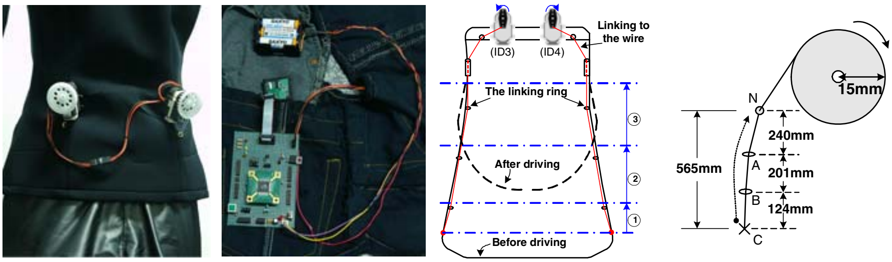


(a) front view (b) back view (c) skeleton view

Fig. 2. Transformation devices mounted on the jacket

The attached motors are used for making some transformation for the cloth, and the controller is used for generating the process. Here, the three motors on the jacket, installed on left-upper, right-lower, and center position, respectively, wound around the clothes. These motors are AIMOTOR-601[11] series from Megarobotics, which consist of a motor, a reduction gear, and a position sensor all in one structure. It can control the angular position of motor shaft through a serial communication. Therefore, this motor is very suitable for a multiple agent system controlling a number of motors. The current position and torque values are monitored during the cloth transformation via UART. In this study, since the motor identification numbers are limited to 5 digits, 31 configuration motors can be controlled.

Figure 3 shows the internal shape on the trouser. Two motors in the back-waist direction of trouser are mounted to a tie and are used for raising the trouser. These two motors are operated simultaneously with wires, from the bottom to top to express the smoothness and beauty. The controller is mounted on the front-upper-side of the jacket not to interfere the wrapping operation. The driving motors are installed on the back of waist and connected with the cloth using wires. When the motors start to work, the wires pull the structure upward and the trouser is wrapped upward.



(a) back view (b) front view (c) skeleton view

Fig. 3. Transformation devices mounted on the trouser

The movable range of the clothes can be calculated by using a simple geometry. Figure 2(c) and Figure 3(c) show the mechanical structures attached on shaft of motor in the clothes, where X symbols mean the sewed point of the clothes on the jacket and trouser and solid lines represent wires from A to M and C to N. The clothes wrapping between point A and C can be calculated from moving difference along the AC line. OC or OA is calculated to 249.12mm from Pythagorean theorem. The wrapping sequence starts from turning a quarter, then length of OA becomes about 20.12mm (249.12-250mm+21mm). Until the motor is turned to a quarter, the wires do not be wrapped at the shaft. If the motor turns the structure about 2 turns, working length becomes 50.27mm ($2\pi r\times 2$). Finally it is followed a quarter turn ($2\pi r/4=$ 6.28mm). Therefore overall amount of transformation is 20+50.27+6.28 = 76.55mm. The other wrapping point B is calculated as 42+50.27 = 92.27mm after a half turn (42mm) and 2 turns (50.27mm). In case of the trouser, there are almost 12 turns, so total length of transformation can be measured ($2\pi \times 7.5 \times 12$ turns = 565.49mm). The lower, middle and upper parts of the attaching points are supposed to fold at the point of 124.49mm, 201mm and 240mm respectively. The purpose of installation of the ring shape elements at point A and B is not to make the cloth transformation until reaching the sewed point C at those points. The above process took 5 seconds (2.5 turns) for the jacket and 12 seconds (6 turns) on the trouser.

3.2 Structure of Main Controller and WSN

Figure 4 shows the main structure of the controller and functional block diagram of the system. It consists of a controller module and a WSN module. Most of basic circuit and software are based on a typical TinyOS. A basic CPU in a controller is ATmega128 with a 128KB flash memory, 4KB RAM, SPI, I2C, and USART, all in one chip. The TinyOS is a kind of sensor network operating systems, which is very easily to install on a small system and has the abilities such as time scheduling and networking. It can be included user application programs like configuration scenarios. The control operation is minimized and managed in real-time since it is event-driven including support of network managements.

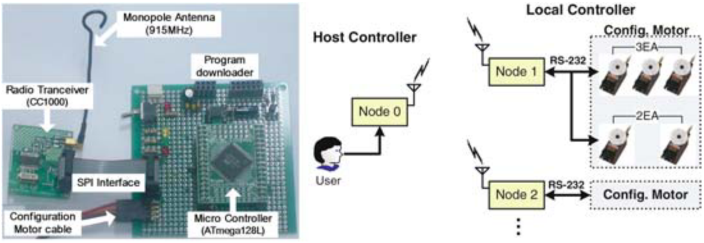
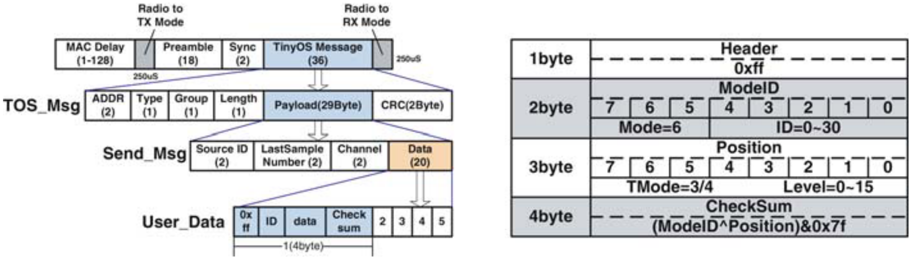


Fig. 4. Main controller and its configuration

The development environment is also very convenient since it uses a general PC program environment like a Cygwin and NesC[12]. The controller has the two functional modules, one is an embedded system with TinyOS and the other is a cloth controller. The controllers also basically contain a WSN with a 915MHz CC1000 chip, so it is easy and reliable to connect to the other controllers through network protocol like Ad-hoc and Multi-hop[13].



1byte	Header 0xff							
2byte	ModelID							
	7	6	5	4	3	2	1	0
	Mode=6			ID=0~30				
3byte	Position							
	7	6	5	4	3	2	1	0
	TMode=3/4			Level=0~15				
4byte	CheckSum (ModelID^Position)&0x7f							

Fig. 5. Wireless packet architecture and motor configuration packets

The information exchange is 28 bytes per 100ms as shown in Figure 5. This data contains the transformation data for the motors. The data packet for controlling the configuration motors is shown in the right of the Figure 5. It is transferred via serial channel using a multi-drop method. All the configuration motors are tied in this channel and followed the communication specification of AIMotor-601. Many commands configure the motor operation, but just position command (Wheel Act Mode) is available to transform clothes. The driving speed levels of motor are divided into 16 steps. Motors named from ID0 to ID3 are used for jacket transformation, here ID0 and ID1 moved to counter clockwise and clockwise for an ID2. Whenever ID0 and ID2 rotate 5 times, ID1 rotates 3 times. On the other hand, there are 2 motors in a trouser, the ID3 moving counterclockwise, and ID4 moving clockwise direction. All sequence takes 12 rotations.

4 Experiment

To verify our study, we carried out an experiment as follows. First, a predefined scenario was downloaded offline to the remote controller before the normal operation.

The remote controller can send the clothes transformation data to the local controller by WSN. They can exchange the control and monitor data with each other after packing or unpacking the communication packets. In this experiment, two clothes, single jacket and single trouser, were controlled by remote user at a show stage. All sequences were synchronized with an internal timer. Figure 6 shows the functional state diagram of the controllers.

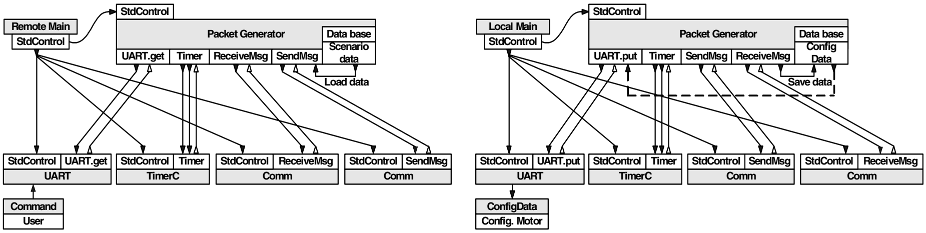


Fig. 6. State diagram for a remote controller and local controller

In the Figure, PacketGenerator module loads the configuration data into communication packets to transmit to a local controller, which is installed in the flash memory of remote controller in advance. On the other hand, PacketExtractor in a local controller draws the configuration data from the packets. This data is transferred to the configuration motor, which carries out the clothes transformation. Whenever the timer expires at a rate of 100msec, it sends 12 operational scenarios, which contain the clothes configuration data. They are used to drive the configuration motors, and its operation will be repeated until deformation cloth is renewed. TinyOS supports many modules. Among them, CC1000RadioM is used for media access control of wireless communication, TimerM supports the 100ms time intervals, and HPLUART0M module manages the serial communication using UART0 with the configuration motors. In this experiment, the Mica2 remote controller from XBOW was used since it does not require any additional hardware, but a local controller was included by the user hardware. These controllers were mounted on the clothes in a jacket and trouser. Figure 7 describes the clothes transformation sequence for the jacket and trouser. The sequence was carried out in 6 steps. In the case of jacket, phase (1) shows the initial posture without transformation, then the left-upper and a right-bottom motors were operated in order during phases (2) to (4); in phase (5), the middle motor started to work to fasten tightly; and finally, in the phase (6), the transformation sequence was finished. In the trouser case, phase (1) is the initial stage, and from phase (2) to phase (6), the two motors at both ends operated in order, and then the trouser was wrapped upward slowly at a rate of tens centimeters at a second. This operation took 5 seconds for the jacket and 12 seconds for the trouser. To verify the configuration process, the real situation in fashion-show was performed. The left of Figure 8 depicts the flow chart of operation process. During a show, the performance manager controls the deformation steps, and monitors the information related with the shape of the clothes. From the remote location in near far away, an operator with a WSN node controls the shape of the clothes at the show spot.

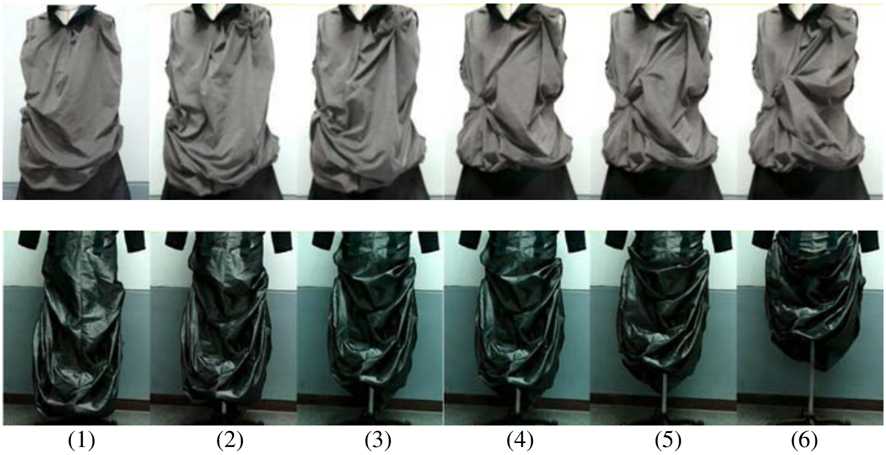


Fig. 7. Configuration process of a jacket (upper) and a trouser (bottom)

The right of Figure 8 shows the real performance with these clothes configurations.

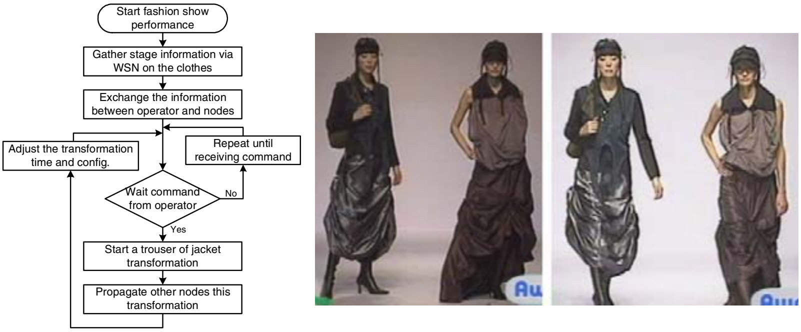

Start fashion show performance
Gather stage information via WSN on the clothes
Exchange the information between operator and nodes
Adjust the transformation time and config.
Repeat until receiving command
Wait command from operator
No
Yes
Start a trouser of jacket transformation
Propagate other nodes this transformation

Fig. 8. The flow chart of the configuration process and real performance

As seen in the figure, the clothes mounted on fashion models were well configured according to command by remote performance manager, and also the multiplicity for the clothes is well expressed by transformation process.

5 Conclusion

This paper proposed a clothes configuration method, which can be used in a place like a fashion-show stage. This method applied WSN having an embedded operating system, and also included the servo motor interface used to transform the shapes of clothes. Therefore, a user from a remote place can carry out various configurations during a show performance. The controller was designed as two modules: one was a remote controller, and the other was a local controller. They were controlled with WSN based on TinyOS from Berkeley. All the hardware and software structure were discussed and presented, and their function in a real performance such as in fashion-show stage was realized actually. The experimental results showed that our proposal was a good method to configure various kinds of design samples at a show stage. In the future, this concept based on networking will be expanded to the multiple clothes configuration based on multi-hop protocol, and a more compact controller is necessary to make the clothes wearer more comfortable.

Acknowledgement

This work was supported by Grant No. R01-2003-000-10364-0 from Korea Science & Engineering Foundation.

References

1. Eun-Young Lee: A Fashion Marketing,Kyomunsa, Korea(1999)
2. Young-Ja Lim and Yoon-Suk Han: A study of Formativeness on the Influence of the Memphis on the Comtemporary Fashion Design, Proceedings of the Korean Society of Costume Conference, Vol. 51, No. 1(2001) 5-20
3. Eun-Young Kim, Kyomunsa: A Fashion material planning and information, Korea(2000)
4. E. H. Callaway: Wireless Sensor Networks: Architectures and Protocols. 2000 N.W. Corporate Blvd., Boca Raton, Florida 33431: CRC Press LLC(2003)
5. http://www.berkeley.edu/news/media/releases/2001/12/13_snsor.html
6. A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson: Wireless sensor networks for habitat monitoring, Atlanta, GA (2002)
7. A. N. Knaian: A wireless sensor network for smart roadbeds and intelligent transportation systems, Ph.D. dissertation, Massachusetts Institute of Technology (1999)
8. Sarah Bergbreiter and K.S.J Pister: CotsBot: An Off-the -Shelf Platform for Distributed Robotics, Proceedings of the 2003 IEEE/RSJ Intl. Congerence on Intelligent Robots and Systems Las Vegas, 1632-1637, Nevada(2003)
9. Gabriel T. Sibley, Moharmmad H. Rahimi and Gaurav S. Sukhatme: Robomote: A Tiny Mobile Robot Platform for Large-Scale Sensor Networks, Proceedings of the IEEE International Conference on Robotics and Automation(2002)
10. Y. Chen, K. Moore, and Z. Song: Diffusion based path planning in mobile actuator-sensor networks (mas-net) - some preliminary results," Int. Proc. of SPIE Conf. on Intelligent Computing: Theory and Applications II, part of SPIEs Defense and Security(2004)
11. http://www.megarobotics.com
12. D. Gay, P. Levis, R. von Behren, M. Welsh, E. Brewer, and D. Culler: The nesC language: A holistic approach to networked embedded systems, ACM SIGPLAN Conference on Programming Language Design and Implementation (2003)
13. Alec Woo, Terence Tong and David Culler: Taming the Underlying Challenges of Reliable Multihop Routing in Sensor Networks, Proceedings of the 1st Int. Conf. on Embedded networked sensor systems (2003),14-27

Hash-Based RFID Security Protocol Using Randomly Key-Changed Identification Procedure

Jia Zhai, Chang Mok Park, and Gi-Nam Wang

Industrial & Information Systems Engineering Department,
Ajou University, Suwon, South Korea
zhaijiaws@vip.sina.com, cmpark25@hanafos.com,
gnwang@ajou.ac.kr

Abstract. Radio Frequency Identification (RFID) is considered to be a promising identification approach in ubiquitous sensing technology. The operation of RFID systems in advanced applications may pose security and privacy risks to both organizations and individuals. In this paper, using randomly Key-Changed Identification, we propose an eavesdropping-proof security protocol based on cryptographic one way hash functions for passive RFID tags. Compared with several existing methods, our proposed protocol shows some security improvements as well as gives a reasonable and compatible approach that could be easily employed in practical situations. Finally, an illustration is also given to show clearly the whole operating procedure of the proposed procedure. Key Words: RFID (Radio Frequency Identification), Security Protocol, Hash Functions, Randomly Key-Changed Identification.

1 Introduction

The Radio Frequency Identification (RFID) technology is an emerging technology that uses radio waves to automatically identify individual items [1][2]. Through automatic and real-time data acquisition, this technology can give a great benefit to various industries by improving the efficiency of their operations [3][4]. It could be applied for checking storage goods in advanced logistics applications such as JIT environments [5]. Much more complicated applications can be found in the areas such as auto-distribution production line, warehouse security moving in and out check, and smart shelves in the shop. Furthermore, it could be also effectively deployed under the extremely hazard circumstances where human can not reach a high temperature production process in a pharmaceutical plant [6]. Compared with the existing bar code system, RFID has lots of special advantages such as it does not require physical communication line, it can be reprogrammed easily, it can be used in harsh environment, it can store more data, and it can read many tags simultaneously. We might anticipate RFID technology will be the next generation solution in object identification and tracking material status under a ubiquitous computing environment.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 296–305, 2006.

A complete RFID system basically consists of three components: transceiver (reader), transponder (tag), and data management infrastructure: the back-end [7]. The transceivers (reader) can read data from and also write data to a transponder. It also works as a power supplier in the passive RFID systems. The transponder, or tag, is usually attached to an object to be identified, and stores data of the item. There are two types of RFID tags: the active one which generates power by itself normally an on-board battery and the passive one which gets energy from the transceiver through radio frequency. The passive tag seems to be much more attractive than the active one because of the price factors, which the preferable price for pervasive deployment of RFID tags would be about $0.05 [7]. The remaining part would be the subsystem usually a data management infrastructure (back-end) bridging the gap between the physical and the digital world. It may contain both the enterprise application layer and the middle layer software such as Savant [8]. These three parts cooperate efficiently to implement all kinds of applications using RFID technology.

2 RFID Security Problems

The RFID security problem becomes extremely crucial along with the drastically increasing deployment of RFID tags. This problem is viewed as a barrier to the widespread adoption of RFID technology. The privacy information leakage and location tracking seems to be the most serious problems in realistic domain.

In everyday life, people do not want others to know what he carried or what he bought at a supermarket because some of them might be quite personal. But due to the trend that RFID tags gradually become unified such as Class I tags defined by EPC [7], it is quite easy to detect what other people carries just by a third party reader. Furthermore, logistics and inventory data hold significant financial value for commercial organizations and their adversaries. If items in the whole supply chain of a company are equipped with RFID devices, it will be extremely dangerous to employ RFID technology in all wide supply chain process without any data encryption method. The adversary of a company can easily obtain the complete manufacturing process information of its corresponding manufacture plant only by interrogating the unprotected tags.

Another important concern is the tracking of individuals by RFID tags. This problem is concerned much after a major tire manufacturer decided put RFID tags into its tire products [9]. Individuals might be tracked by the tires used in their personal cars. Except this, other goods labeled by tags like clothes, shoes might also expose location information of individuals who wear them in their daily life.

However, the inborn properties of RFID devices, such as low-power, low computing ability, storage resource-deficiency, and low-price for pervasive deployment make it quite difficult to apply a perfect data protection method such as strong symmetric or even asymmetric encryption method in large scale applications. For keeping the price of passive tag under $0.05, a practical design of passive tag only contains roughly 7.5k - 15k gates. Because it requires 5k to 10k gates for a typical 100 bit EPC chip, only approximately 2.5k - 5k gates are available for security control in passive RFID tags [10]. As a result, all the data encryption solutions for RFID system try to find a good balance between the cost of hardware and the ability of data protection.

In this paper, based on randomly key-changed identification procedure, we present a security protocol to prevent the eavesdroppers from monitoring the communication data between RFID devices by using cryptographic one way hash functions on the passive tags. In the following sections, we will look over some related research works firstly. Then, we propose our eavesdropping-proof solution using randomly key-changed identification scheme. An operation example is also shown for giving illustration of the complete procedure. Finally, the conclusion and further studies are also discussed.

3 Related Works

There are several existing works dealing with security problems in RFID system, we discuss some important ones here.

In Sanjay E. Sarma's research work [7], a Hash-Based Access Control procedure is presented. To lock a tag, the database computes a hash value of a random key and sends this hashed value to the tag as a metaID while saving the original keys in the database. In turn, the tag stores the metaID and enters the locked status. While in the locked status, a tag responds to all queries only by the metaID. After the database received the metaID value, it hashes all the keys stored and finds which matches the metaID then sends the matched key back to the tag. The tag hashes this value and compares it to the metaID. If the value matches, the tag unlocks itself. This scheme is relatively low cost because only a hash function is needed on the tags. However, it is easy to be middle-attacked since one can query tags for getting the metaID value and send it to the reader, then later unlocks the tag with the reader's responding matched key.

M. Ohkubo also presented a security scheme that specially considered the forward security in RFID system [10]. This scheme employed two hash functions and a hash chain to renew the important information contained on the tags. But this scheme is still vulnerable to the middle attack with the reason same as the above Hash-Bashed Access Control procedure [7]. In our proposed method, this kind of attack could be avoided by randomly changing the interrogating key between tags and reader.

The Kill-Command method was developed by Auto-ID Center [11]. This method enables RFID tags destroying information stored on it when necessary. Although it can protect the information detected by other parties or individuals, this method actually cuts off many important benefits of RFID systems. Moreover, someone might be able to prevent the tags destroying themselves in order to obtain the secret information.

Ari. Juels proposed an External Re-encryption Algorithm [12]. Public-key encryption of serial numbers in RFID tags is employed in this algorithm. However, the difficulty of this algorithm is the data stored on each tag have to be updated often. This limitation makes it difficult to operate this scheme in practice [10]. On contrary, our method only needs updating several bits in passive tags so that our system is much easier to be deployed in practical situations.

In Stephen A. Weis's work [13], he gave a Randomized Access Control method which tried to improve the Hash-Based Access Control method. However, this method requires passive tags employing a pseudo-random generator. As mentioned above, only 2.5k - 5k gates could be used for security control in passive tags, so this kind of method might not be embedded at a low cost in practical domain because the minimal hardware complexity of a PRF ensemble remains an open problem [14]. Compared with this method, our proposed scheme does not need a PRF generator inside the passive tags which might be more reasonable in real applications.

There are also some other works bear to mention such as Avoine's paper [15]. By using Time-memory Trade-off, they gave an advanced algorithm based on M. Ohkubo's work [10] for adding the scalable ability. Furthermore, a different security aspect known as 'yoking-proof' is discussed by Saito's work [16]. They solved the replay attack by using a time stamp.

Although the existing methods gave some security protocols dealing with the privacy issues for RFID system, they have some flaws which might stop them from being deployed in large scale practical applications. In [7][10][15], the middle attacking threaten could not be avoided. By killing the tags themselves, the method shown in [11] may also kill many inherit advantageous of RFID technology. Moreover, [12]'s scheme requires rewriting data stored on each tag frequently, [13]'s algorithm needs a PRF generator inside passive tags; so these two methods might be difficult in practical implementations. An attempt is given to improve the previous works by presenting our randomly key-changed identification scheme, which possibly the previous improper points could be corrected or improved by our approach.

4 Problem Definitions

To address our proposed procedure, we would like give some assumptions firstly. We assume the communication layer between readers and back-end is not exposed to the eavesdroppers. It means that the eavesdroppers can only listen to the communication signals between tags and readers and can not get any information concerning the reader to back-end communication layer. This is illustrated by Figure 1.

Furthermore, we assume each tag contains two statuses: locked and unlocked. In the locked status, the tags only send back the Meta-Key for any enquiries, whereas in the unlocked status, the tags offer all functionalities to the reader. At the beginning, all the tags are initiated as the locked status.

Because the signal sent from tags to reader is relatively much weak, and it may only be monitored by eavesdroppers within the tag's shorter operating range, we also generally assume that in the unlocked status, eavesdroppers can not listen to the functional signals directly without any detection. On contrary,

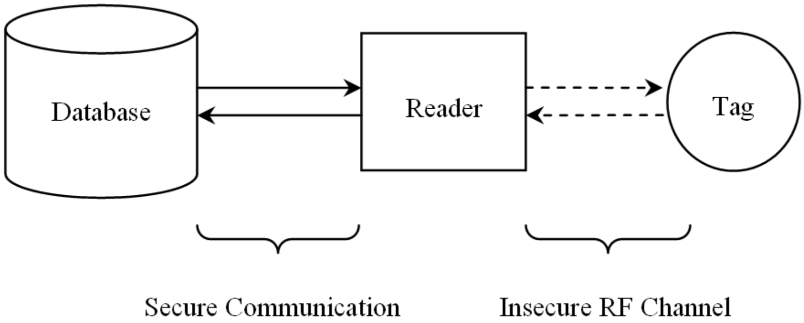


Fig. 1. Compared with the insecure RF channel between tags and reader, the communication layer between back-end and reader is secure enough

during the locked status, the eavesdroppers may monitor the Meta-Key sent by the tags. Thus the key point of our eavesdropping-proof scheme is to safely change the status of locked tags without being detected by eavesdroppers and also prevent the eavesdroppers from unlocking the tags illegally. The illustration of this assumption is shown as Figure 2.

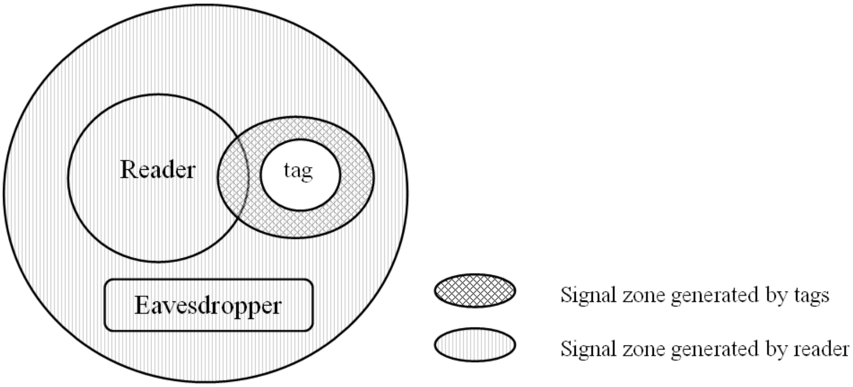

Reader
tag
Eavesdropper
Signal zone generated by tags
Signal zone generated by reader

Fig. 2. An illustration of eavesdropping scene in the unlocked status. Generally without detection, the eavesdropper can not monitor the functional communication data from tags to reader.

5 Randomly Key-Changed Identification Procedure

5.1 Initial Setup

The main idea of our proposed method is to keep the interrogation key between readers and tags changing randomly during the runtime of RFID system. We give two functions as preliminaries below:

$$Y = h(X) \tag{1}$$

$$Y = g(X) \tag{2}$$

These two functions are cryptographic one-way hash functions.

Furthermore, both the back-end and tags need to contain some bits information used in our scheme.

Key: "K" Stored both in back-end and tags.

Meta-Key: "MK" Stored only in back-end.

Verifying-Key: "VK" Dynamically generated.

Flag-Key: "FK" Stored both in back-end and tags.

To lock a tag with an ID value at the beginning, the back-end generates a random value A as well as computes Y=h(ID) by (1). Then we let:

$$K = ID \tag{3}$$

$$MK = Y \tag{4}$$

$$FK = A \tag{5}$$

The set value (K, MK, FK) are saved in the back-end system while the set value (K, FK) are saved in the tag, then this tag enters its locked status and responds any enquiries only by MK value. For each new tag, we use this procedure for initialization.

5.2 Randomly Key-Changed Identification Procedure

In this part, we would like introduce our proposed scheme. When we need the functionalities of a tag, the reader broadcasts the query signal firstly. After receiving the query signal, tag hashes K value by (1) and sends back MK=h(K) to the reader. Then the back-end also uses (1): Y=h(K) hashing all the K value stored in the database and finds which Y matches MK so that we can get a set of value (K, MK, FK) which is related to the received MK.

Then the back-end generates a random value B and sends (VK, FK, B) back to the tag where VK=h(K||B). Using K value it stored and the received B value, the tag computes Y=h(K||B). If Y matches received VK as well as FK value matches, the tag unlocks itself. Finally, both the database system and the tag refresh the FK value by (2):

$$FK = g(K||B) \tag{6}$$

The completed state diagram is shown in Figure 3.

Before the unlocking action, all the tags check carefully if the FK values matched. Since the FK value is a stochastic value: FK=g(K||B) that keeps changing at each interrogation cycle, our algorithm could resolve the middle-attack efficiently while other similar existing methods employed a PRFs (pseudo-random functions) generator inside the passive tags such as [13] which might be not practical [14].

As we discussed before, the complexity of a security protocol for RFID system is extremely important as the low cost requirement for passive tags. Our scheme requires computation of two hash functions: (1) and (2). Because the cryptographic one way hash functions embedded on the tags and managing keys on the back-end might achieve a relatively low cost [10][17], our proposed method could be well deployed in a large application scale while other methods may not.

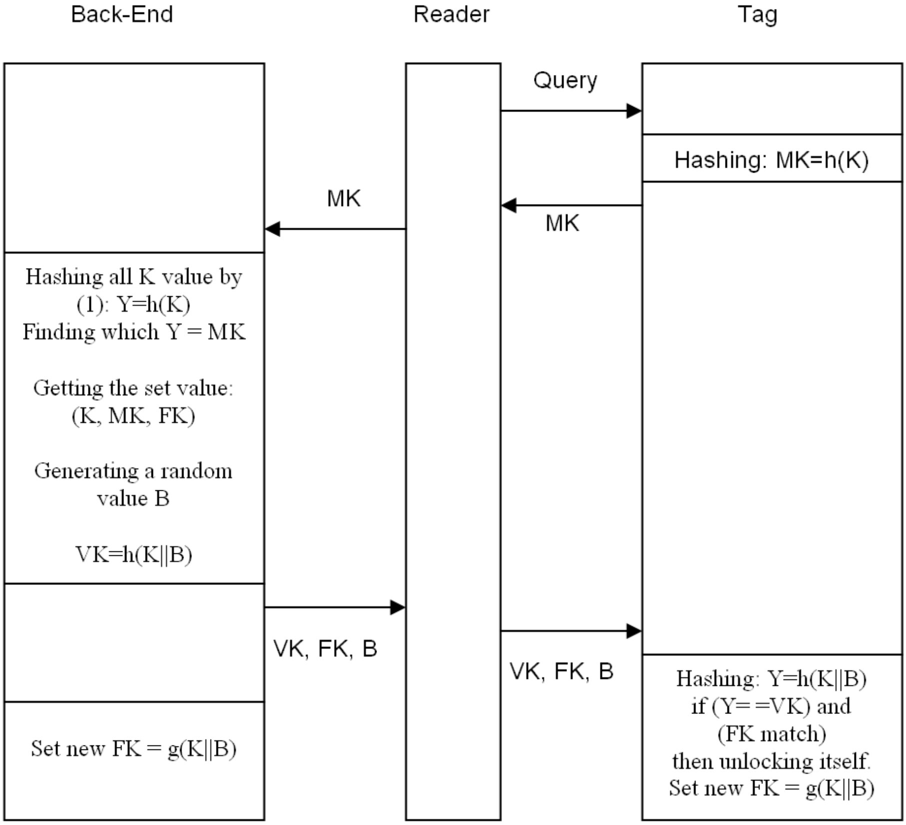


Fig. 3. The state diagram of our proposed identification procedure using randomly key-changed interrogation

6 Illustration for the Scheme

For making the operation of our protocol clear and demonstrating how the proposed method works against the middle attack, an example of completed operation is shown in this section.

We assume that there is a new tag with ID value 10. Before this tag enters its locked status, the back-end should initiate the system. Firstly, the back-end generates a random number A, let it just be 7, and hashes ID value by (1): Y=h(ID), here ID value is 10. So we can get three values by (3) (4) (5):

$$K = ID = 10 \tag{7}$$

$$MK = Y = h(10) \tag{8}$$

$$FK = A = 7 \tag{9}$$

Then, those data will be stored both in the back-end and this tag shown as Table 1 and Table 2, and this tag enter its locked status:

Table 1. Data stored in the back-end at the beginning

K	MK	FK	DATA
10	h(10)	7	Reference

Table 2. Data stored in this tag at the beginning

K	FK	DATA
10	7	other EPC code

During its locked status, this tag responses all kinds of queries just by MK value h(10). For common eavesdroppers, they can not get any information only from the MK value h(10), so the basic security assurance is validated. When the back-end wants to unlock this tag, it hashes all the K values it has by (1): Y=h(K) until one result Y equals h(10) so that we could get a set of (K, MK, FK) as (10, h(10), 7).

At present, the back-end needs generating another random value B and we assume B is 8. Then the reader sends (h(10||8), 7, 8) as (VK, FK, B) to the tag. After receiving this three values, this tag:

(1) Hashing stored K and received B by (1) : Y=h(K||B)=h(10||8).

(2) Because VK=Y=h(10||8) and FK matches, this tag unlocks itself.

(3) Both the back-end and the tag sets new FK=g(10||8).

Therefore, the data stored in the back-end and this tag now is shown as Table 3 and Table 4. As well as the functionality communication finishing, this tag will enter its locked status again.

Table 3. Data stored in the back-end after one functional reading cycle

K	MK	FK	DATA
10	h(10)	g(10||8)	Reference

Table 4. Data stored in this tag after one functional reading cycle

K	FK	DATA
10	g(10||8)	other EPC code

For a middle attacker, he may get the response signal from the tag, and tries to use this signal getting information from the reader, then unlocks the tag some time later by pretending to be a legal reader. This kind of attack can not be prevented in several existing RFID security protocols like [7][10][15]. In our example, the middle attacker could get (h(10||8), 7, 8) from the reader by using tag's responding MK value h(10). However, after finishing this functionality reading cycle, both the FK stored in the back-end and this tag have already been changed by a new random value: FK=g(K||B)=g(10||8). Hence, when the middle attacker tries to unlock the tag some time later by using signals (h(10||8), 7, 8) with FK=7, the unlocking attempt will be denied by the tag because the FK value does not match. By employing this scheme, we successfully make sure the important information stored on the tag keeping secure against the middle attackers.

7 Conclusion

We have demonstrated a security protocol for identifying passive RFID tags safely. Our proposed randomly key-changed identification procedure could be employed in many kinds of applications without relying on strong symmetric or even asymmetric encryption. The main advantage of the proposed method is its simplicity because it only requires hash functions on the tag and data management at the back-end. Only simple message exchange is required, the radio communication channel need not be reliable, and the reader need not be trusted. The proposed scheme is well compatible to the existing EPC standard because only simple EPC codes need storing on the tag while complex product information stored in the back-end as references. Furthermore, proved by the illustrative example, middle attacking problem has been successfully resolved by our scheme which could not be prevented in several existing works. As a further study, application specific algorithms can be developed based on our research work, which might facilitate the pervasive deployment of RFID system.

References

1. Jia Zhai and Gi-Nam Wang: An Anti-collision Algorithm Using Two-Functioned Estimation for RFID Tags. ICCSA 2005, LNCS 3483. (2005) 702-711
2. Klaus Finkenzeller: RFID Handbook: Fundamentals and Applications in Contactless Smart Card and Identification. Second Edition. John Wiley & Sons. (2002) 221-228
3. Bornhovd C., Tao Lin, Haller S.; Schaper J.: Integrating Smart Items with Business Processes An Experience Report. System Sciences, 2005. HICSS '05. Proceedings of the 38th Annual Hawaii International Conference on , 03-06 Jan. (2005) 227c-227c
4. Geng Yang, Jarvenpaa S.L.: Trust and Radio Frequency Identification (RFID) Adoption within an Alliance. System Sciences, 2005. HICSS '05. Proceedings of the 38th Annual Hawaii International Conference on , 03-06 Jan. (2005) 208a-208a
5. L. Csera, J. CseleAnyib, M. Geigerc, M. MaEntylaEd, A.S. Korhonene: Logistics from IMS towards virtual factory. Journal of Materials Processing Technology 103 (2000) 6-13
6. Susy d'Hont: The Cutting Edge of RFID Technology and Applications for Manufacturing and Distribution. Texas Instrument TIRIS (2002)
7. Sanjay E. Sarma, Stephen A. Weis, and Daniel W. Engels: RFID Systems and Security and Privacy Implications. CHES 2002, LNCS 2523 (2003) 454-469
8. Clark S., Traub K., Anarkat D., Osinski T.: Auto-ID Savant Specification 1.0. Auto-ID Center, White Paper MIT-AUTOID-TM-003, Sep. (2003)
9. Stephen A. Weis, Sanjay E. Sarma, Ronald L. Rivest and Daniel W. Engels: Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems. Security in Pervasive Computing 2003, LNCS 2802 (2004) 201-212

10. M. Ohkubo, K. Suzuki, and S. Kinoshita: Cryptographic approach to "privacy-friendly" tags. In RFID Privacy Workshop, MIT, USA, (2003)
11. Auto-ID Center, "860MHz-960MHz Class I Radio Frequency Identification Tag Radio Frequency & Logical communication Interface Specification Proposed Recommendation Version 1.0.0", Technical Report MIT-AUTOID-TR-007, Nov. (2002)
12. Ari. Juels, Ravikanth. Pappu: Squealing euros: Privacy protection in RFID-enabled bank-notes. In proceedings of Financial Cryptography - FC' 03, LNCS 2742 (2003) 103-121
13. Stephen A. Weis, Sanjay E. Sarma, Ronald L. Rivest and Daniel W. Engels: Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems. Security in Pervasive Computing 2003, LNCS 2802 (2004) 201-212
14. Matthias Krause and Stefan Lucks: On the Minimal Hardware Complexity of Pseudorandom Function Generators, In Theoretical Aspects of Computer Science, volume 2010, LNCS (2001) 419-435
15. Avoine, G.; Oechslin, P. : A Scalable and Provably Secure Hash-Based RFID Protocal. Pervasive Computing and Communications Workshops, 2005. PerCom 2005 Workshops. Third IEEE International Conference on. (2005) 110-114
16. Saito, J.; Sakurai, K.; : Grouping proof for RFID tags. Advanced Information Networking and Applications, 2005. AINA 2005. 19th International Conference on Volume 2, 25-30 March (2005) 621-624
17. Henrici, D.; Muller, P.: Hash-based Enhancement of Location Privacy for Radio-Frequency Identification Devices using Varying Identifiers. Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on 14-17 March (2004) 149-153

Counting-Based Distance Estimations and Localizations in Wireless Sensor Networks

Oh-Heum Kwon and Ha-Joo Song

Division of Electronics, Computer, and Telecommunication Engineering,
Pukyong National University, Pusan, Rep. of Korea
ohkwn@pknu.ac.kr

Abstract. We present two distributed algorithms for localizing sensor nodes of a wireless sensor network. Our algorithms determine locations of nodes based on the connectivity between nodes. The basic idea behind our algorithms is to estimate distances between nearby nodes by counting their common neighbors. We analyze the performance of our algorithms experimentally. The results of experiments show that our algorithms achieve performance improvements upon the existing algorithms.

1 Introduction

Wireless sensor networks (WSN) are becoming increasingly important in a large number of applications. The basic concept is to deploy a large number of low-cost sensor nodes in a region. Each node monitors its environment, processes the sensed data, and transmits the result to the central system. In many applications of WSNs, the sensed data is useful only when considered in the context of where the data was taken from. In addition, the location information of nodes is a useful primitive in some routing protocols [6], information dissemination protocols [4], and sensor query processing systems [8]. Therefore, the *localization* of the sensor nodes is considered to be a fundamental problem in WSNs.

An obvious solution is to equip each node with a Global Positioning System (GPS) receiver. But using GPS requires significant increases in size, cost, and energy consumption, and then, is unsuitable for low-cost sensor networks. Thus there is interest in developing methods to determine locations of the sensors with a minimum of additional hardware. A number of localization algorithms that do not depend on GPS have been proposed. Those algorithms can be classified into two categories: the *physical measurement-based* approaches and the *connectivity-based* approaches.

In the *physical measurement-based* approaches, each node *physically* measures distances to neighboring nodes and/or angles from which the signals arrive. The positions of nodes are determined based on the measured distances or angles. Typical measurement methods include RSSI (Received Signal Strength Indicator), ToA (Time of Arrival), TDoA (Time Difference of Arrival), and Angle of Arrival (AOA) [3, 11, 12, 13, 14].

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 306–315, 2006.

In the *connectivity-based* approaches, the positions of nodes are determined based only on the connectivity relationships between nodes. Two nodes are considered to be connected if they can communicate directly. In those approaches, it is typically assumed that each node has a fixed communication range and can communicate directly with the nodes within the range. Then the connectivity relationships between nodes are represented as a graph, called the *unit disk graph*. The positions of nodes are determined by considering the graph-theoretic properties of the unit disk graph like hop-counts, degrees, and the existence of subgraphs of certain topologies [1, 9, 10].

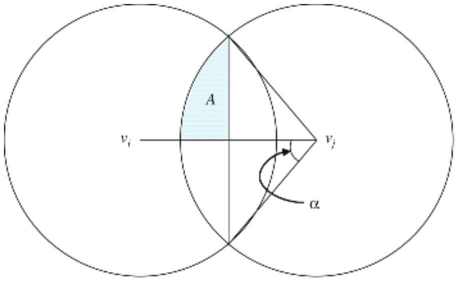
In general, we can expect a higher accuracy of localization with the physical measurement-based approaches. The connectivity-based approaches, however, have an obvious benefit in respect of cost-effectiveness. Location accuracy requirements vary with the applications. There are many applications of WSN in which the cost-effectiveness is more crucial than the accuracy of the localization. In this paper, we study the connectivity-based localization approaches.

Most of the existing connectivity-based distributed localization algorithms share a common structure [5]. It is assumed that there are at least three special nodes, called *anchors*, that know their positions *a priori*. Those nodes get the location information by manual initialization or an additional positioning system like GPS. The localization algorithms consist of three phases: (1) each node estimates distances from each of the anchors; (2) each non-anchor node determines its position using the procedure called *multilateration*; and (3) the determined positions are adjusted by an iterative *refinement* procedure.

Multilateration is a common tool for locating a node using predetermined positions of three or more anchors and the estimated distances from those anchors [10, 14]. Suppose that node v knows coordinates of $k \geq 3$ anchors (x_i, y_i), $1 \leq i \leq k$, and the estimated distances d_i, $1 \leq i \leq k$, from node v to each of the anchors. Then the coordinate (x, y) of node v is determined to the one that minimizes the total squared error between the calculated distances and the estimated distances, that is, $\sum_{i=1}^{k}(\sqrt{(x_i - x)^2 + (y_i - x)^2} - d_i)^2$.

The *DV-hop* algorithm [9] and the *Amorphous* algorithm [10] are the representative connectivity-based localization algorithms. In those algorithms, each node first determines how many hops away from each of the anchors. In order to convert the hop-counts to distances, they introduce the concept of *average hop distance*, and then estimate distances to the hop-counts multiplied by the average hop distance. The *DV-hop* algorithm defines the average hop distance as the sum of the actual distances between anchors divided by the sum of the hop counts between anchors, while the *Amorphous* algorithm [10] defines it as a function of the node density using the Kleinrock and Sliverster formula [7].

Another connectivity-based localization algorithm, called GHoST [1], takes into account the topological properties of the unit disk graph. The algorithm investigates certain local structures, called trimmers and stretches, to drive a lower and/or upper bound on the inter-node distances. Those bounds help the algorithm to avoid some worst cases of distance estimations.



$$\alpha = \arccos \frac{d_{ij}}{2r}$$

$$A = \pi r^2 \frac{\alpha}{2\pi} - B$$

$$B = \frac{1}{2} r \cos \alpha \cdot r \sin \alpha$$

$$common\_area = 4A$$

$$\rho_2 = \frac{common\_area}{\pi r^2} \text{ and } \rho_1 = \frac{|N_i \cap N_j|}{|N_i|}$$

Fig. 1. Calculating common area

In this paper, we assume that every node has the same communication range r. Although this assumption may not be a realistic description of wireless sensor networks in physical environments, it is a valid starting point for modelling purposes, and has been studied by various groups in the past [2, 10].

The basic idea behind our algorithms is to estimate distance between two nearby nodes by counting their common neighbors. In general, the closer two nodes are, the higher the number of the common neighbors will be. For two nodes u and v that have common neighbors, we assume that the number of common neighbors is proportional to the area of the intersection of two disks of radius r centered at u and v, respectively. Upon this assumption, it's possible to guess the distance between two nodes.

Based on the counting-based distance estimations, we construct two localization algorithms, called *Trigonometric* and *Progress_Estimation*, respectively. They have the common structure and achieve significant performance improvements upon the existing algorithms.

2 Distances Between at Most Two Hop Apart Nodes

Let us introduce some notations that will be used throughout this paper. We are given n sensors with the same communication range r. Let d_{ij} denote the physical distance between sensor nodes v_i and v_j. Let $G = (V, E)$ be the unit disk graph. Each node $v_i \in V$ represents a sensor node and each edge $(v_i, v_j) \in E$ represents the fact that the distance between sensors v_i and v_j is at most r, that is, $d_{ij} \leq r$. The existence of edge (v_i, v_j) means that direct bidirectional communication is possible between two nodes. Let h_{ij} denote the hop-count between v_i and v_j in graph G. The hop-count is the minimum number of edges included in the path connecting two nodes in the graph. A node v_j is called a *neighbor* of v_i if $h_{ij} = 1$. For any node v_i, let N_i be the set of neighbors of v_i in G.

Suppose that node v_j is at most two hops apart from node v_i, that is, $h_{ij} \leq 2$. Then they have common neighbors and the distance d_{ij} is at most $2r$. We assume that the number of common neighbors is proportional to the area of the intersection of two disks of radius r centered at v_i and v_j. Let ρ_1 be the ratio of the number of common neighbors of v_i and v_j to the number of neighbors of v_i

and let ρ_2 be the ratio of the common area of two disks to the area of the disk centered at v_i. Note that ρ_1 can be computed if we know $|N_i|$ and $|N_i \cap N_j|$, while ρ_2 can be expressed as a function of distance d_{ij}, as shown in Figure 1.

We estimate the distance d_{ij} by solving the equation $\rho_1 = \rho_2$. Actually, the equation can be approximately solved by applying binary search on d_{ij}. The search range is $[0, r]$ if v_j is a neighbor of v_i; $[r, 2r]$ if v_j is two hop away from v_i. At most 5 iterations of the binary search suffice for the desired accuracy of our use.

3 Localization Algorithms

The framework of our localization algorithms is as follows: (1) each node finds its neighbors, two-hop apart nodes, and the common neighbors with each of them; (2) each node estimates distances from each of the anchors; (3) each node determines its coordinate through the multilateration.

3.1 Finding Neighbors and Common Neighbors

In this subsection, we describe how to find the neighbors, the two-hop apart nodes, and the common neighbors with each of them. Two types of messages are used in this phase. Each node first broadcasts a message notifying its existence to its neighbors. We call this message the *notifying message*. The message includes the sender's id. If any node v_i receives a notifying message from its neighbor v_j, node v_i adds its own id into the received message and broadcasts it to its neighbors again. We call this message the *relayed message*. For this relayed message, we call node v_j the *originator* and node v_i the *relayer*. Each notifying message is relayed just once by each of its neighbors. The relayed messages are not relayed anymore. So the existence of a node is propagated two-hops across the network.

Upon the completion of this phase, each node v_i can find neighbors, two-hop apart nodes, and the common neighbors with each of them: if node v_i receives the notifying message of node v_j, then v_j is a neighbor of v_i; if v_i receives the relayed message with originator v_k but does not receive the notifying message of v_k, then v_k is two-hop apart from v_i. For both cases, the number of relayed messages with originator v_j is the number of common neighbors with v_j.

Let's count the number of total messages involved. Each node broadcasts one notifying message and the notifying message is relayed once by each of its neighbors. Therefore, the number of messages originated from a node is the degree of the node plus 1. Summing up this for every node, the number of total messages is $n + 2m$, where n is the number of nodes and m is the number of edges of the unit disk graph.

3.2 Estimating Distances from Anchors

In what follows, we describe how to estimate distances from a specific anchor node v_0 to other nodes. In this section, we use the shortened notations d_i and h_i

instead of d_{0i} and h_{0i}, respectively. Our algorithm is similar to the well-known distance vector routing algorithm. During the execution of the algorithm, each node discovers its hop-count from the anchor v_0 and estimates its distance from the anchor. Each message includes the anchor v_0's coordinate and the sender's hop count.

The anchor v_0 initiates the algorithm by broadcasting a messages to its neighbors. If any node v_i receives a message from its neighbor v_j, node v_i examines the sender's hop-count h_j included in the message. If either h_i is not defined yet or $h_i > h_j + 1$, node v_i accepts the message and sets its hop-count h_i to $h_j + 1$; otherwise, it rejects the message. If node v_i accepts the message, it estimates its distance d_i. We explain how to estimate d_i later in this section. Then, node v_i broadcasts a new message to its neighbors. For this new message, we call node v_i the *sender* of the message, and call node v_j the *predecessor* of the message. The message includes the following information:

1. the id and the coordinate of anchor v_0
2. the id of sender v_i, its hop-count h_i, and the estimate of distance d_i
3. the id of predecessor v_j and the estimate of distance d_j
4. the estimate of distance d_{ij}

The initial message from anchor v_0 to its neighbors, of course, does not have the predecessor, and hence, the message does not have item (3) and (4). In general, the items (1) and (3) are available from the prior message sent by the predecessor v_j, and item (4) can be estimated by the sender v_i. How to estimate distance d_i is the topic of the remaining part of this section.

Suppose that node v_i has just received a message. If $h_i \leq 2$, node v_i can estimate its distance d_i using the method described in Section 2. Consider the general case of $h_i > 2$. Suppose that v_{i-1} is the sender and v_{i-2} is the predecessor of the message that v_i has just received. Node v_i has the estimates of the following five distances:

$$d_{i-1},\ d_{i-2},\ d_{i,i-1},\ d_{i-1,i-2},\ \text{and}\ d_{i,i-2}$$

The estimates of three distances d_{i-1}, d_{i-2}, and $d_{i-1,i-2}$ are available from the message received, and the other two can be computed by node v_i using the method in Section 2. In what follows, we propose two distinct methods, called *Trigonometric* and *Progress_Estimation*, to estimate d_i using them.

3.3 *Trigonometric* Method

See Figure 2. Three nodes v_i, v_{i-1}, and v_{i-2} form a triangle, and three nodes v_0, v_{i-1} and v_{i-2} also form another triangle. There are two possible positions of node v_i satisfying five distance constraints, as shown in Figure 2. The one is the reflection of the other against the mirroring line connecting v_{i-1} and v_{i-2}.

Choosing the correct one among two candidate positions is a kind of problem called the *folding problem* [11]. The folding problem must be handled very carefully when a very accurate distance measurement technique like TDoA is used and a high degree of location accuracy is desired. But it's doubtful to be worth handling when the distance estimations are erroneous like ours.

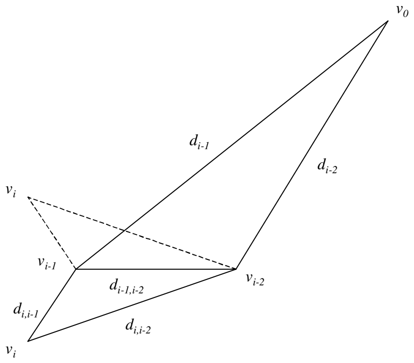


Fig. 2. Trigonometric Method

To clarify this issue, we implement and compare two methods: the one that always chooses the farthest one from v_0 among two candidates and the other that always chooses the correct one. Choosing correct one means that the decision is made based on the real coordinates of four nodes v_i, v_{i-1}, v_{i-2}, and v_0. More specifically, we made the decision by examining whether v_i and v_0 are actually in the same side of the mirroring line or not. The experiments exhibit no noticeable difference in performance between two. It means that the error caused by the wrong selections is overwhelmed by the distance estimation error. So we take the simplest strategy that always chooses the farthest one from v_0. Calculating the length of the diagonal $\overline{v_i v_0}$ is just an arithmetic, and the description will be skipped.

3.4 *Progress_Estimation* Method

In this method, node v_i tries to guess the displacement $\Delta_i = d_i - d_{i-1}$ rather than d_i itself. Once Δ_i is appropriately guessed, distance d_i is estimated to $d_{i-1} + \Delta_i$.

The intuition behind this method is very simple. Suppose that there is a very twisty path. We will come close to the destination a little by moving one hop along the path, in general. If the path is almost straight, on the other hand, we will come close to the destination almost as much as the hop distance by one hop.

See Figure 3. Using three estimates of distances $d_{i,i-1}$, $d_{i-1,i-2}$, and $d_{i,i-2}$, it's possible to know how much bending at node v_{i-1} the min hop path from v_i to v_0 is. Let α_{i-1} be the acute angle $\angle v_i v_{i-1} v_{i-2}$. Applying cosine rule, the angle α_{i-1} can be computed as shown in Figure 3.

Let q be the intersection point of the straight line connecting v_i and v_0 and the circle of radius d_{i-1} centered at v_0. Then, $\Delta_i = |\overline{v_i q}|$. If v_0 is sufficiently far away from v_i, then $|\overline{v_i q}| \simeq d_{i,i-1} \cdot \cos\theta_i$ where θ_i is the acute angle $\angle v_{i-1} v_i v_0$. If we impose an appropriate probability distribution on θ_i, the expectation $E[\Delta_i]$ can be expressed as follows:

$$E[\Delta_i] \simeq \int_0^{\pi} \text{prob}(\theta_i = \theta) \cdot d_{i,i-1} \cos\theta \; d\theta, \qquad (1)$$

where prob$(\theta_i = \theta)$ is the probability of θ_i being θ.

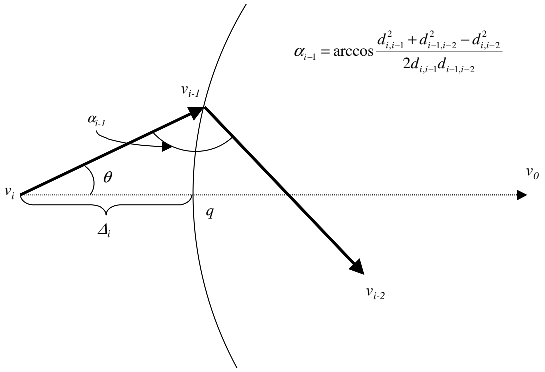

$\alpha_{i-1} = \arccos \frac{d_{i,i-1}^2 + d_{i-1,i-2}^2 - d_{i,i-2}^2}{2d_{i,i-1}d_{i-1,i-2}}$
v_{i-1}
α_{i-1}
θ
v_0
v_i
q
Δ_i
v_{i-2}

Fig. 3. Three consecutive nodes

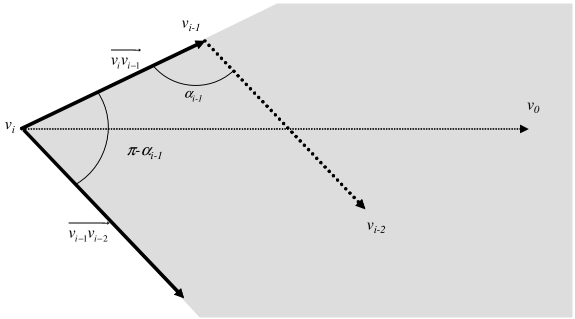

v_{i-1}
$\overrightarrow{v_i v_{i-1}}$
α_{i-1}
v_0
v_i
π-α_{i-1}
v_{i-2}
$\overrightarrow{v_{i-1} v_{i-2}}$

Fig. 4. Assumption on the position of v_0

Analyzing prob($\theta_i = \theta$) for randomly distributed nodes seems to be challenging, but difficult. Instead of analyzing that, in this paper, we take a very optimistic and simple assumption on prob($\theta_i = \theta$). See Figure 4. We simply assume that anchor v_0 is in direction between two vectors $\overrightarrow{v_i v_{i-1}}$ and $\overrightarrow{v_{i-1} v_{i-2}}$ originated at node v_i, which means that $0 \leq \theta_i \leq \pi - \alpha_{i-1}$. In other words, node v_0 is assumed to be somewhere within the shaded area of Figure 5. Moreover, we assume that the probability of θ_i being θ, for $0 \leq \theta \leq \pi - \alpha_{i-1}$, is equally likely. In summary, we assume that

$$\text{prob}(\theta_i = \theta) = \begin{cases} 0 & \text{if } \theta > \pi - \alpha_{i-1}, \\ \frac{1}{\pi - \alpha_{i-1}} & \text{if } 0 \leq \theta \leq \pi - \alpha_{i-1}. \end{cases} \tag{2}$$

Substituting equation (2) into (1), we have

$$E[\Delta_i] \simeq \frac{1}{\pi - \alpha_{i-1}} \int_0^{\pi - \alpha_{i-1}} d_{i,i-1} \cdot \cos\theta \; d\theta = \frac{\sin \alpha_{i-1}}{\pi - \alpha_{i-1}} \cdot d_{i,i-1}$$

In summary, node v_i estimates its distance d_i as follows:

$$d_i = \begin{cases} \text{estimated by the method in Section 2} & \text{if } h_i \leq 2, \\ d_{i-1} + \frac{\sin \alpha_{i-1}}{\pi - \alpha_{i-1}} \cdot d_{i,i-1} & \text{if } h_i > 2. \end{cases}$$

4 Simulation Results

In this section, we present the results of our simulations. We do not include the *Amorphous* and the *GHoST* algorithm in our presentation of the results. The *Amorphous* algorithm exhibits a little worse results than the *DV-hop* algorithm for almost all settings we've tested. We believe that the *GHoST* algorithm is a sort of technique that can be added to any algorithm to enhance the performance.

In our simulations, the nodes are randomly placed according to a uniform distribution on a squared area of size 500 × 500 units. The number of node is 250. The specified fraction of anchors is randomly selected from the nodes.

Figure 5 shows the error of anchor-to-node distance estimations. The results show that *Progress_Estimation* performs best. *Trigonometric* is worse than *Progress_Estimation* slightly and consistently.

A noticeable phenomenon is that the distance estimation accuracy of *DV-hop* decreases as the node density increases over a turning point. It's natural since increasing density while preserving the number of nodes means enlarging the communication range r. Enlarging the range, the hop-counts between nodes are getting smaller and the granularity of distances becomes coarser.

Figure 6 show the localization error rates for anchor percentage 3% and 8%, respectively. The error rates are measured by the location errors divided by range r. The results are as expected from the distance estimation error rates. The *Progress_Estimation* performs best and achieves 30% of the location error at the average degree about 12.

5 Concluding Remarks

A weakness of our distance estimation method is that it is based on an ideal model of the communication range. In practice, communications between nodes within the communication range can fail by various and unpredictable reasons. The communication range itself can also vary from node to node according to the configurations of the ground.

Nevertheless, we think our method has several advantages and possibilities for practical use. First, it requires no additional hardware support. Second, the distance estimation part of our algorithms does not require even the existence of anchors, which gives us a potential to use our technique in *anchor-free* localizations. Third, the counting-based distance estimation method can be used in the refinement phase of other connectivity-based localization algorithms like *DV-hop*. In typical refinement techniques, each node iteratively updates its

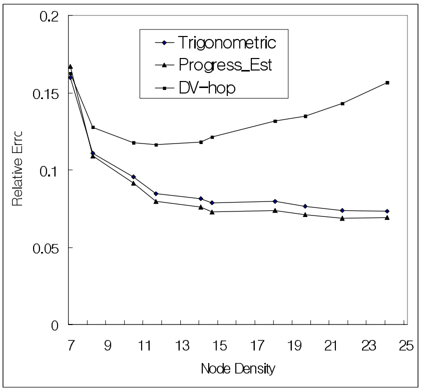


Fig. 5. Anchor-to-node distance estimation error

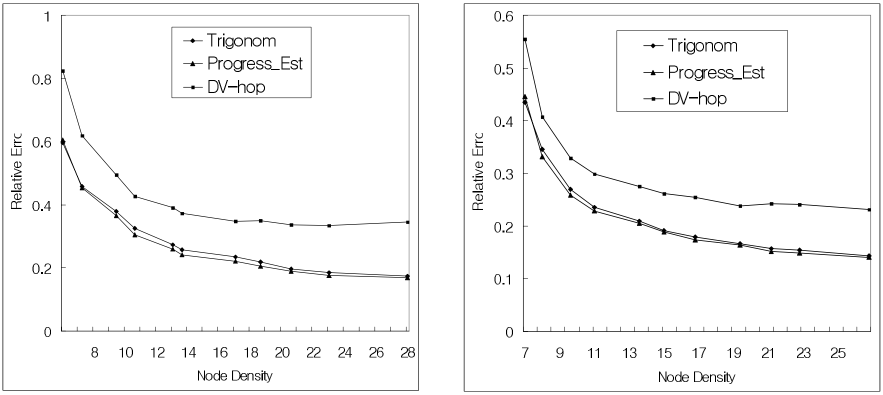


Fig. 6. Relative location error with 3% anchors and 8% anchors

coordinate by considering the coordinates of the surrounding nodes and the distances from them. In those contexts, the relative magnitudes of the distances are more important than the absolute magnitudes of the distances, which can make the weakness of our technique mentioned above less fatal.

References

1. R. Bischoff and R. Wattenhofer, “Analyzing Connectivity-Based MultiHop Ad-Hoc Positioning,” Proc. of the Second Annual IEEE International Conference on Pervasive Computing and Communications, 2004.
2. A. Galstyan,B. Krishnamachari, K. Lerman, and S. Pattem, “Distributed online localization in sensor networks using a moving target,” The Third Symposium on Information Processing in Sensor Networks, IPSN’04, 2004.

3. T. He, C. Huang, B. Blum, J. Stankovic, and T. Abdelzaher, “Range-free localization schemes in large scale sensor networks,” Proceedings of The Ninth International Conference on Mobile Computing and Networking (Mobicom), pp. 81-95, San Diego, CA, Sep 2003.
4. C. Intanagonwiwat, R. Govindan, and D. Estrin, “Directed diffusion: A scalable and robust communication paradigm for sensor networks,” Proc. of the Sixth International ACM Conference on Mobile Computing and Networking (MOBICOM), Boston, MA, 2000, pp. 56-67.
5. K. Langendoen and N. Reijers, “Distributed localization in wireless sensor networks: a quantitive comparison,” Computer Networks, Vol. 43, pp. 499-518, 2003.
6. J. Li, J. Jannotti, D. De Couto, D. Karger, and R. Morris, “A scalable location service for geographic ad-hoc routing,” Proc. 6th ACM MOBICOM Conf, Boston, MA, Aug. 2000.
7. L. Kleinrock and J. Silverster, “Optimum transmission radii for packet radio networks or why six is a magic number,” Proc. Natnl. Telecomm. Conf., pp 4.3.1-4.3.5, 1978.
8. S. Madden, M. Franklin, J. Hellerstein, and W. Hong, “TAG: a tiny agregation service for ad-hoc sensor networks,” Proc. of the Fifth USENIX Symposium on Operating Systems Design and Implementation (OSDI), Boston, MA, December 2002.
9. D. Niculescu and B. Nath, “DV based positioning in ad hoc networks,” Telecommunication Systems, Vol. 22:1-4, pp. 267-280, 2003.
10. R. Nagpal, H. Shrobe, and J. Bachrach. “Organizing a Global Coordinate System from Local Information on an Ad Hoc Sensor Network,” 2nd International Workshop on Information Processing in Sensor Networks (IPSN 03), Palo Alto, CA, Apr. 22-23, 2003.
11. N. B. Priynatha, H. Balakrishnan, E. Demaine, and S. Teller, “Anchor-free distributed localization in sensor networks,” MIT Laboratory for Computer Science, Technical Report No. 892, April 15, 2003.
12. N. Patwari and A. O. Hero III, “Using proximity and quantized RSS for sensor localization in wireless networks,” WSNA’03, September 19, 2003, San Diego, California, USA.
13. N.Patwari,A.O.H.III,M.Perkins,N.S.Correal, and R. J. O’Dea, “Relative location estimation in wireless sensor networks,” IEEE Trans. on Signal Processing, Vol. 51, No. 8, 2003, pp. 2137-2148.
14. Andreas Savvides, Chih-Chieh Han, and Mani B. Srivastava, “Dynamic fine-grained localization in ad-hoc networks of sensors,” Proceedings of the Seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2001), Rome, Italy, July 2001.

Self Re-encryption Protocol Providing Strong Privacy for Low Cost RFID System*

Jeong Su Park, Su Mi Lee, Eun Young Choi, and Dong Hoon Lee

Center for the Information Security Technologies(CIST),
Korea University, 1, 5-Ka, Anam-dong, Sungbuk-ku, Seoul, 136-701, Korea
{standalon, donghlee}@korea.ac.kr, {smlee, bluecey}@cist.korea.ac.kr

Abstract. RFID(Radio Frequency Identification) system will play a critical role providing widespread services in a ubiquitous environment. However, widespread use of RFID tags may create new threats to a consumer privacy such as information leakage and traceability. It is difficult to solve the problems because a tag has the limited computing power that lacks of supporting the general encryption. Although the scheme of [2] protects a consumer privacy using an external agent, a tag should perform exponential operation requiring high cost. We propose *Self Re-Encryption Protocol*(SREP) which provides strong privacy without help of any external agent. SREP is well suitable for low cost RFID systems since it only needs multiplication and exclusive-or operation.

1 Introduction

RFID system is the automatic identification technology and can rapidly read the information of an object without physical contact. Since the system is capable of wireless communication over a short distance, it will become the substitution of bar-code system. Nowadays, the system is partially used for supply chain management, traffic management, animal management, and so on.

In wireless communication with an RFID tag, an RFID reader obtains its information and transmits the information to a database. Service providers can easily obtain the consumer information included in a tag anywhere and provide the consumer with various convenient services using the information. However, an adversary can eavesdrop the information because the message is transmitted over an insecure channel. For this reason, it is possible to reveal the sensitive consumer information such as credit information, health state, shopping patterns, and so on. Also, the adversary can trace the location of the consumer holding the tag by analyzing the eavesdropped message.

One of the most important issues in an RFID scheme is a computational cost of a tag since the tags[1] have an extremely limited computing power. Researchers

* This research was supported by grant No.R01 − 2004 − 000 − 10704 − 0 from the Korea Science & Engineering Foundation.

[1] RFID tag is divided into an active tag and a passive tag. In this paper, we consider all tags as the passive tag. The features of the passive tag are limited computing power, low cost, power supply by induction, and so on.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 316–325, 2006.

have suggested numerous protocols to reduce the computational cost and to protect the consumer privacy. As a result, there are various approaches such as a physical approach and a cryptographic approach which is classified into a hash function approach and a re-encryption approach. In this paper, we solve privacy problems by using the re-encryption approach. Namely, whenever a reader transmits a request to a tag, the tag always re-encrypts its information for itself and transmits the new ciphertext. Therefore, an adversary cannot infringe the consumer privacy. In addition, since the tag of our scheme requires the only low cost operations(multiplication and exclusive-or), our scheme is suitable for low cost RFID systems.

1.1 Related Work

Researchers have recognized the privacy problems of an RFID system and devised various approaches to protect the consumer privacy. In the initial paper, the scheme uses the physical approach such as *Kill Command*[11], *Blocker Tag*[3], *Faraday Cage*[5], and *Active Jamming*[3]. In the cryptographic approach, the authentication scheme using a hash function is *Hash-Lock*[9, 10, 11], *Hash-Chain*[6, 7], *Challenge-Response Authentication scheme*[4], and so on.

The other cryptographic approach, *re-encryption approach*, is the method that is to re-encrypt the value of a tag's information periodically using a asymmetric key[1] or a symmetric key[8] by an external agent. The schemes of [1] and [8] prevent information leakage by encryption and traceability by re-encryption. Since the computing power of a tag is so low that the tag cannot perform the re-encryption requiring exponential operation, an external agent re-encrypts the tag's information instead. Since the tag always emits the fixed response until the next re-encryption step, it should re-encrypt its information with the external agent in every session.

In *Universal Re-encryption scheme*[8], an external agent re-encrypts information without knowledge of public keys. Saito *et al.*[2] showed attack models that an adversary who impersonates an external agent writes particular value to an target tag in re-encryption step. In order to prevent the vulnerabilities, the tag of *Re-encryption with a check* checks if the re-encrypted ciphertext is correct before writing the ciphertext to the tag. However, the tag should perform the exponential operation.

1.2 Contribution

We propose the scheme which is applied to low cost RFID systems and protects the consumer privacy. In this section, we express contributions of efficiency and privacy.

- **Efficiency.** As explained above, it is important to reduce the computational cost of a tag in RFID system. The scheme of [2] is inadequate in low cost RFID systems because the tag performs exponential operation for re-encryption. However, since the tag of our scheme only performs multiplication and exclusive-or(XOR) operation, it is possible to apply our scheme to low cost RFID systems.

- **Privacy.** In [1, 2], there are many vulnerabilities in a tag by using the unauthenticated external agent. If an adversary prevents a target tag from re-encrypting its information with an external agent, the tag always transmits the same ciphertext whenever a reader requests. The adversary can trace the location of the consumer holding the tag through fixed ciphertext. However, an tag of our scheme always performs re-encryption for itself without help of an external agent and emits the new ciphertext to a reader in every session. Therefore our scheme provides stronger privacy than the schemes [1, 2].

Organization. The remainder of this paper is as follows : We explain privacy problems in section 2. In section 3, we describe characteristics or roles of each component. Then in section 4, we explain SREP in detail, and analyze the SREP in view of security and efficiency in section 5. We conclude in section 6. In appendix, we show that SREP is secure against replay attack and spoofing attack.

2 Privacy Problems

In an RFID system, there are various vulnerabilities such as eavesdropping, analysis and alteration of the transmitted information. An adversary tries to disclose the sensitive information, trace the location of a consumer holding a tag, and so infringe the consumer privacy. In this section, we do not consider about replay attack and spoofing attack. We will explain that our scheme applying challenge-response technique is secure against the attacks in appendix.

- **Information Leakage.** In a ubiquitous environment, most people will have several tags including various information which is very personal and sensitive. For example, many people may not want to disclose the name of medicine, a expensive product and the title of a book. Since the tag's information is transmitted over an insecure channel, the sensitive information will be easily disclosed by an adversary without the knowledge of tag's holder.
- **Traceability.** If an adversary knows whether the transmitted message is the information of the target tag or not, he can trace the location of a consumer holding the target tag. Although the adversary cannot understand a message, if a link between the target tag and the message is established, he also trace the location of the consumer.

3 RFID Components

In our scheme, an administrator divides all tags into several groups and several databases manage the groups. We add a new component that is called *Gateway* to classify the transmitted message to the corresponding database. Therefore, we newly define RFID components which are adequate in our scheme. RFID system of our scheme consists of *RFID Member Tag*(member tag), *RFID*

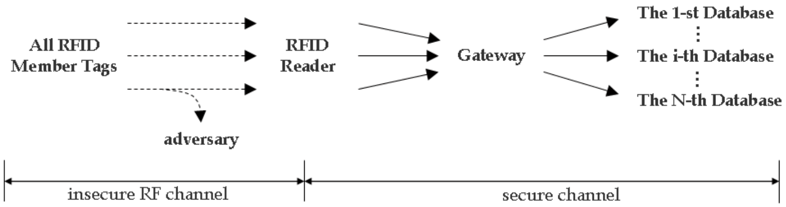


Fig. 1. The overview of RFID system

Reader(reader), gateway and *Group Database*.[2] The features or the roles of each component are as follows.

- **RFID Member Tag:** This component is an inexpensive microchip that has the limited computing power. It emits its information in response to a request from a nearby reader.
- **RFID Reader:** This component includes an antenna and a microchip for wireless communication with a member tag. It only gathers information from nearby member tags through an RF channel and transmits the information to gateway. It has a high computing power in comparison with a member tag.
- **Gateway:** This component has a high computing power. It computes a transmitted message with its key and transmits the message to all group databases.
- **Group Database:** In our scheme, we assume that each group database manages the information of its member tags and its private key securely. Upon receiving a message from a gateway, the group database decrypts the transmitted ciphertext by its private key and obtains information of a member tag.

4 Self Re-Encryption Protocol (SREP)

SREP consists of the setup phase and the execution phase. We assume that an administrator securely performs the setup phase. When a reader transmits a request to a member tag, the execution phase is performed.

4.1 Setup

- **STEP 1** (Administrator) **Grouping**
  An administrator randomly divides all member tags into several groups. The number of a group(N) should be adequately determined by considering security and efficiency.

[2] We change the name of an RFID tag and a database to emphasize the meaning of a group in this paper. The characteristics of the components are equal to that of the general RFID components except the name.

- **STEP 2** (Administrator) **Key Generation**
  Let $\mathcal{G}$ denote the underlying group for ElGamal cryptosystem, let q denote the order of $\mathcal{G}$, and let g and s be a published generator for $\mathcal{G}$[8]. The administrator generates four keys for every group as following Table 1.[3]

Table 1. Keys of each group

Key	Symbol	Expression
Group Private Key	x_i	$x_i \in_R \mathbb{Z}_{q-1}$
Group Public Key	y_i	$y_i = g^{x_i}$
Update Public Key	u_i	$u_i = s^{x_i}$
Gateway Symmetric Key	GS_i	$GS_i \in_R \mathbb{Z}_{q-1}$

- **STEP 3** (Administrator) **Encryption**
  Let $m_{i,j}$ denote a member tag's information.[4] An administrator chooses new random numbers k_0, k_1, k_2 and k_3 for every member tag. He calculates ciphertext $C_{i,j}$ and *Update value* $UD_{i,j}$ using the random numbers as following formula (1) and writes $C_{i,j}$, $UD_{i,j}$ and GS_i to a member tag securely. He securely stores all GS_i to gateway and x_i to the i-th group database each.

$$\begin{aligned} C_{i,j} &= [(\alpha_0, \beta_0); (\alpha_1, \beta_1)] = [(m_{i,j} y_i^{k_0}, g^{k_0}); (y_i^{k_1}, g^{k_1})] \\ UD_{i,j} &= [(\gamma_0, \delta_0); (\gamma_1, \delta_1)] = [(u_i^{k_2}, s^{k_2}); (u_i^{k_3}, s^{k_3})] \end{aligned} \tag{1}$$

Briefly, we describe secret values stored in each component of our scheme as following Table 2.

4.2 Execution

- **STEP 1** (RFID Member Tag) **Self Re-Encryption**
  A member tag re-encrypts its ciphertext $C_{i,j}$ into new re-encrypted ciphertext $C'_{i,j}$ as following formula (2).

$$\begin{aligned} C_{i,j} &= [(\alpha_0, \beta_0); (\alpha_1, \beta_1)] \\ UD_{i,j} &= [(\gamma_0, \delta_0); (\gamma_1, \delta_1)] \\ &\Downarrow \\ C'_{i,j} &= [(\alpha'_0, \beta'_0); (\alpha'_1, \beta'_1)] \\ &= [(\alpha_0 \alpha_1 \gamma_0, \beta_0 \beta_1 \delta_0); (\alpha_1 \gamma_0 \gamma_1, \beta_1 \delta_0 \delta_1)] \end{aligned} \tag{2}$$

- **STEP 2** (RFID Member Tag) **Group Symmetric Encryption**
  The member tag encrypts the ciphertext $C'_{i,j}$ into a ciphertext $P_{i,j}$ using GS_i as following formula (3) and transmits the ciphertext $P_{i,j}$ to a gateway through a reader.

$$P_{i,j} = C'_{i,j} \oplus GS_i \tag{3}$$

[3] Key_i denotes the key of the i-th group.

[4] $\text{Info}_{i,j}$ denotes the information that the j-th member tag of the i-th group stores.

Table 2. The secret values of each component

Component	j-th Member Tag	i-th Group	Group Symmetric key	Update value	Ciphertext	Group Private Key
All RFID Member Tags	tag$_3$	group$_1$	GS_1	$UD_{1,3}$	$C_{1,3}$	-
	tag$_{11}$			$UD_{1,11}$	$C_{1,11}$	
	$\cdots$			$\cdots$	$\cdots$	
	tag$_4$			$UD_{1,4}$	$C_{1,4}$	
	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	-
	tag$_{97}$	group$_N$	GS_N	$UD_{N,97}$	$C_{N,97}$	-
	tag$_n$			$UD_{N,n}$	$C_{N,n}$	
	$\cdots$			$\cdots$	$\cdots$	
	tag$_{36}$			$UD_{N,36}$	$C_{N,36}$	
Reader	-	-	-	-	-	-
Gateway	-	-	GS_1 $\cdots$ GS_N	-	-	-
Group Database	-	group$_1$	-	-	-	x_1
	-	group$_2$	-	-	-	x_2
	-	$\cdots$	-	-	-	$\cdots$
	-	group$_N$	-	-	-	x_N

– **STEP 3** (Gateway) **Gateway Transmission**
As shown in the following Fig.2, a gateway calculates all GS_i to ciphertext P with XOR operation and transmits the message to each group database.

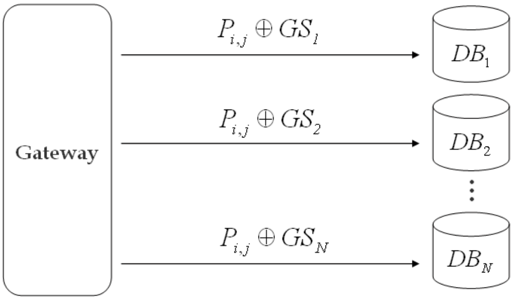


Fig. 2. Transmission of Gateway

– **STEP 4** (Group Database) **Decryption**
After performing decryption algorithm Table 3, the only group database of the transmitted message obtains the member tag's information $m_{i,j}$.

Table 3. Decryption Algorithm in Group Database

```
compute $m_0 = \alpha_0/\beta_0^{x_i}$, $m_1 = \alpha_1/\beta_1^{x_i}$
If ($m_1 == 1$)
    information $m_{i,j} \leftarrow m_0$
else
    reject
```

5 Analysis

5.1 Security

In communication of between a member tag and a reader, a passive adversary eavesdrops all ciphertexts P, and an active adversary analyzes, alters the ciphertexts or impersonates an authenticated reader. We analyze the security of our scheme against the threats introduced in section 2.

An adversary tries to find out secret values of a target member tag that are ciphertext C', UD and GS. The adversary can obtain ciphertexts P and calculated values using the ciphertexts P, which is presented in following Table 4[5].

Table 4. The forms of message that an adversary can use

Ciphertexts P disclosed over an insecure channel	Calculated values using the ciphertexts P
$P^1 = C' \oplus GS$	$P^1 \oplus P^2 = C' \oplus C''$
$P^2 = C'' \oplus GS$	$P^1 \oplus P^3 = C' \oplus C^{'''}$
$P^3 = C^{'''} \oplus GS$	$C' \oplus C'' \oplus C^{'''} \oplus GS$
$P^4 = C^{''''} \oplus GS$	$C' \oplus C'' \oplus C^{'''} \oplus C^{''''}$
$\cdots$	$\cdots$

If an adversary find out the only one value among the secret values of a member tag, she can compute the other secret values by analyzing ciphertexts P of the member tag. As shown in Table 4, she cannot obtain the secret values independently, but the forms of the secret values associated with XOR operation (for example, $C' \oplus GS$, $C' \oplus C''$, $C' \oplus C'' \oplus C^{'''} \oplus GS$, and so on). Therefore, she cannot obtain any secret values of the member tag.

- **Information Leakage.** Although an adversary obtain a ciphertext C' of a target member tag, she cannot obtain the target member tag's information m. Since our scheme uses public key cryptosystem, the only group database holding private key x is able to obtain m. Therefore, our scheme is secure against information leakage.

[5] P^i denotes the transmitted ciphertext P in the i-th session.

- **Traceability.** Although GS is fixed, a ciphertext P is always changed because ciphertext C' is re-encrypted in self re-encryption step in every session. Even if an adversary obtains all re-encrypted ciphertexts P, she cannot decide whether the two different responses are generated by the same member tag. Therefore, our scheme prevents traceability of the target consumer.

Since our scheme is secure against information leakage and traceability, it protects the consumer privacy.

It is a vulnerable point that an unauthenticated external agent writes some value to a member tag. However, our scheme fundamentally solves the potential problems occurred by writing of an external agent, because a member tag of our scheme performs re-encryption step without the external agent for itself.

5.2 Efficiency

It is important to minimize the computational cost of a member tag.

- **The Computational Cost in an RFID Member Tag.** Upon receiving a request from a reader, a member tag performs self re-encryption and group symmetric encryption steps, and transmits a ciphertext P to the reader. As explained in section 4, a member tag performs 8 times multiplication operation in the self re-encryption step and 1 time XOR operation in the group symmetric encryption step.

In paper [2], a member tag requires an exponential operation to check an validity of a ciphertext re-encrypted by an external agent, but it is impossible for the member tag to perform the exponential operation. However, the member tag of our scheme requires multiplication and XOR operation. Therefore, our scheme is more efficient than the re-encryption scheme of [2].

Table 5. The analysis of efficiency

Scheme	The computational cost
Scheme of [2]	Exponential
SREP	Multiplication

6 Conclusion

We propose a scheme that protects the consumer privacy using limited computing power of a member tag. Since SREP executes self re-encryption without an external agent, a member tag always emits the re-encrypted ciphertext on every request of a reader. Therefore, we solve the problems of information leakage and traceability.

We propose practical model for low cost RFID systems. Therefore we expect that our scheme is applied to user belongings such as transportation card and money in near future.

References

1. A. Juels and R.Pappu. *Squealing Euros : Privacy Protection in RFID-enabled Banknotes.* FC'03, vol.2742 of LNCS, pp.103-121, Jan 2003.
2. J. Saito, J. Ryou and K. Sakurai. *Enhancing Privacy of Universal Re-encryption Scheme for RFID Tags.* EUC 2004, vol.3207 of LNCS, pp.879-890, Aug 2004.
3. A. Juels, R. L. Rivest and M. Szudlo. *The Blocker Tag: Selective Blocking of RFID tags for Consumer Privacy.* ACM CCS, pp.103-111, Oct 2003.
4. K. Rhee, J. Kwak, S. Kim and D. Won. *Challenge-Response Based RFID Authentication Protocol for Distributed Database Environment.* SPC 2005, vol.3450 of LNCS, pp.70-84, Apr 2005.
5. mCloak : Personal/corporate management of wireless devices and technology, 2003. http://www.mogilecloak.com.
6. M. Ohkubo, K. Suzuki, S. Kinoshita. *Cryptographic Approach to 'Privacy-Friendly' Tags.* In RFID Privacy Workshop, Nov 2003.
7. M. Ohkubo, K. Suzuki, S. Kinoshita. *Efficient Hash-Chain Based RFID Privacy Protection Scheme.* Ubcomp04 workshop, Sep 2004.
8. P. Golle, M. Jakobsson, A. Juels, and P. Syversion. *Universal re-encryption for mixnets.* CT-RSA, vol.2964 of LNCS, pp.163-178, Feb 2004.
9. S. A. Weis. *Security and Privacy in Radio-Frequency Identification Devices.* Master Thesis, MIT, May 2003.
10. S. E. Sarma, S. A. Weis and D. W. Engels. *RFID systems and security and privacy implications.* CHES02, vol.2523 of LNCS, pp.454-469, Aug 2002.
11. S. A. Weis, S. E. Sarma, R. L. Rivest, and D. W. Engels. *Security and Privacy Aspects of Low-Cost Radio Frequency Identification Systems.* SPC 2003, vol.2802 of LNCS, pp. 454-469, Mar 2003.

A Secure SREP Against Replay and Spoofing Attacks

In this paper, we have not considered two attacks such as replay attack and spoofing attack because we can easily solve the problems by authenticating mutually between a member tag and a group database. In this section, we explain that our scheme applying challenge-response technique is secure against the replay attack and the spoofing attack.

1. A reader transmits a request with a random value R to a member tag.
2. The member tag re-encrypts a ciphertext C into a ciphertext C' in self re-encryption step as explained in section 4.2. It computes $H = h(C' \parallel R)$, where h is an one-way hash function. Then it computes $P = C' \oplus GS$ in group symmetric encryption step.
3. The member tag transmits the ciphertext P to the reader with H_L which is the left half of H. The reader transmits P, H_L and R to gateway.
4. The gateway transmits $C'(= P \oplus GS), H_L$ and R to a group database[6].
5. The group database computes $H' = h(C' \parallel R)$ using the received values. It authenticates the member tag by checking whether the received value H_L equals to H'_L which is the left half of H'.

[6] In this section, we consider the group database according to a transmitted message.

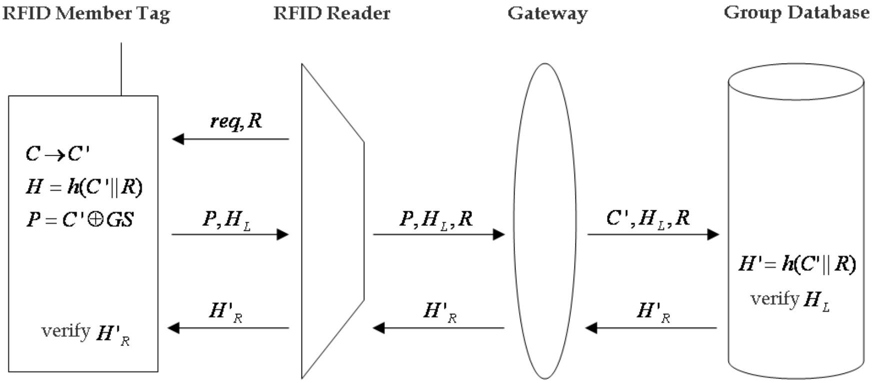


Fig. 3. SREP applying challenge-response technique

6. The database transmits H'_R, which is the right half of H', to the member tag through the reader and the gateway. Upon receiving the value, the member tag authenticates the group database by checking whether the received value H'_R equals to the right half of H.

Although an adversary eavesdrops all transmitted messages between a reader and a member tag, she does not know a ciphertext C' since it is changed in every session. Therefore, the adversary can not compute a hash value H.

Briefly, our scheme applying challenge-response technique is secure against the replay attack and the spoofing attack by authenticating mutually between a group database and a member tag.

Authentication for Single/Multi Domain in Ubiquitous Computing Using Attribute Certification[*]

Deok-Gyu Lee, Seo-Il Kang, Dae-Hee Seo, and Im-Yeong Lee

Division of Information Technology Engineering, Soonchunhyang University, # 646,
Eupnae-ri, Shinchang-myun, Asan-si, Choongchungnam-do, Korea
{hbrhcdbr, pjswise, imylee}@sch.ac.kr
http://sec-cse.sch.ac.kr

Abstract. The Ubiquitous computer environment is thing which invisible computer that is not shown linked mutually through network so that user may use computer always is been pervasive. Intend computing environment that can use easily as user wants and it is the smart environment that user provides context awareness that is wanting computing environment. This Ubiquitous computing contains much specially weak side in security. Masquerade attack of that crawl that is quoted to user or server among device that is around user by that discrete various computing devices exist everywhere among them become possible. Hereupon, in this paper, proposed method that has following characteristic. Present authentication model through transfer or device. Suggest two method that realize authentication through device in case of moved to method (MD: Multi Domain) and user ownself space (SD: Single Domain) that realize authentication through device in case of moved user's direct path who device differs.

1 Introduction

Ubiquitous computing aims at an environment in which invisible computers interconnected via the network exist. In this way, computers are smart enough to provide a user with context awareness, thus allowing the user to use the computers in the desired way. Ubiquitous computing has the following features: Firstly, a variety of distributed computing devices exist for specific users. Secondly, computing devices that are uninterruptedly connected via the network exist. Thirdly, a user sees only the personalized interface because the environment is invisible to him. Lastly, the environment exists in a real world space and not in a virtual one[1][2]. However, the ubiquitous environment is weak in security. Since distributed computing devices are spread out in the environment, it is possible to launch disguised attacks against the environment from a device authenticated by a user or a server. Also, although a user approves of the installation of only authenticated applications into the devices, there is a chance that a malicious code will be transmitted to surrounding devices that do not

[*] This research was supported by the MIC (Ministry of Information and Communica-tion), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 326 – 335, 2006.

have computing capability. Since many ubiquitous computing devices do not provide efficient memory protection, the memory where user information (authentication information) is stored can easily be attacked. These problems can be resolved by using encryption or electronic signature for computing devices. But non-computing devices cannot be protected using encryption codes or electronic signatures, which bring up potential security issues. Also, when a device is moved out of the domain into a new user space, user authentication must be performed smoothly in the new space. This is so because a different device in the new user's space can be authenticated with the user authentication information in the new user space, and not with the previous user authentication information[3][4][7][9]. It is this paper's purpose to propose an authentication model through the movement of a smart device. The conceptualized model proposes two methods. One method is to implement the authentication through a device when an individual small device is moved into the multi domain. The other method is to implement the authentication through a device when another device is moved into the single domain. In the first method, each device stores a different set of user authentication information. Minimum authentication information is stored in the smart device, and most authentication information is stored in the computing device. Since a smart device stores minimum authentication information, it transmits and receives authentication information from the Hub which first provides it with the authentication information. In this method, the Hub can be a user or a server[8]. This paper thus consists of five chapters. Chapter 1 covers this introduction. Chapter 2 covers requirements for authentication in ubiquitous computing. Chapter 3 covers ubiquitous computing research trends. Chapter 4 covers authentication methods using a smart device in the multi domain and the single domain. Chapter 5 compares existing ubiquitous researches and the proposed methods to verify the efficiency of the proposed methods. Finally, Chapter 6 concludes the research[10][11][12][13].

2 Ubiquitous Computing Research Trends and Summary of PMI

In 1993, Mark Weiser of PARC (Palo Alto Research Center), a prominent figure in ubiquitous computing, took a look at the computer evolution in his paper and saw computer technology development from the point of view of changes in human relationships. The first wave was defined as mainframes, each shared by a lot of people. The second wave was defined as personal computers, each shared by one person. In his paper, the ubiquitous society was introduced. In society, a wide range of people use computers without being aware of various internal computers. This computer technology was defined as the third wave. His paper mentions a calm technology that permeates our daily life through invisible interfaces, which will be our major computer interface technology. He predicted that this technological innovation would bring ubiquitous computing to life [4][5][14][15][16][17]. Ubiquitous computing emphasizes the following researches. Since the ubiquitous computing network connects PCs, AV equipment on a server-oriented network, information electronic appliances, cellular phones, game consoles, control devices, and various other devices, core technologies such as miniaturization, cellular phone technology, technology for information electronic appliances, electronic control technology, and networking control technology have risen to the front. Among these, individual authentication technology and security technology have been named as the technologies that will allow users to utilize computers in a secure way. The research on authentication has been conducted as a national project in many countries. However, no research has been done regarding the provision of authentication in the multi-domain. The next part will cover general trends in the ubiquitous computing environment, and will describe existing methods and projects.

2.1 JARM Scheme

In 2002, Jalal's proposed a method that supports the user authentication level concept[3]. Different levels of user authentication information can be stored in different devices, which mean that minimum user information can even be stored in watches and smart rings. Medium-level user information can also be stored in a smart device like a PDA. With this method, if a device is moved from one user domain to another, the device can use the new user information in the new domain. However, the device cannot use the authentication information of the new domain, which restricts users who move from one domain to another from using the device. Therefore, with this method, all devices in one domain have authentication information, and a user can be authenticated through a device and can be authenticated against all devices using the level authentication information. This method cites multiple steps when it comes to the authentication through trust values for level authentication information. A device obtains a trust value by using the authentication protocol suitable for each device. The method that authenticates devices through trust values provides efficient authentication to a smart device, but the method often requires a high-level device to confirm the entire authentication or the smart authentication. If a middle-level device or a high-level device above the smart device is lost or located elsewhere, the entire authentication becomes impossible, thus requiring the redistribution of trust values to devices below that which was lost.

3 Requirements for Ubiquitous Computing

With the advent of human-oriented ubiquitous computing, which is described as pervasive, or invisible computing, a user can concentrate on tasks without being aware that he is using computers. Despite the many benefits of the digital technology that ubiquitous computing utilizes, ubiquitous computing has unseen problems. Without addressing these problems, ubiquitous computing cannot be applied. Since a user uses many devices, user information can be copied in large volume and can be transmitted to unauthorized devices. This illegitimately collected user information can be used maliciously after changes on the network. These features and the environment of ubiquitous computing have allowed for a wide range of malicious attacks and uses, which are likely to become huge obstacles to the development of ubiquitous computing. Thus, to overcome these problems, the following requirements must be met when designing the ubiquitous computing system.

- Mobility: A user's smart device that contains the authentication information must be mobile and be used for all services.
- Entity Authentication: Even when a user with SM_A moves away from Domain_A, the user must be authenticated using the information of SM_A in Domain_B.
- Corresponding Entity Authentication: When Device_B is located in Domain_A, the corresponding entity authentication verifies that Device_B and B are identical entities. This method implements the authentication for devices through the previous user's entity when several devices are connected to one domain. This method can provide a wide range of protection functions.
- Data Outgoing Authentication: When the outgoing data authentication is provided by Domain_A, Domain_A can confirm that Device_A is the actual device in Domain_B that requests the outgoing data authentication. This authentication method provides proof for the authentication data origin. However, this method does not provide protection for data duplication or alteration.
- Connection/Non-connection Confidentiality: Device_B in Domain_A must provide connection confidentiality for the user data. Domain_A receives B's information to obtain the final authentication from the higher-level device. Non-connection confidentiality means that Device_B must provide confidentiality for the user data prior to the connection to a specific domain.

4 Proposed Scheme

In the previous chapters, we have gone over the existing ubiquitous environment, JARM method, and PMI. Although many researches have been done regarding ubiquitous computing, the most active area of research is on communication rather than on security. Security is often researched only as part of the project, not as the main research topic. This paper has selected the JARM method as its research topic since the JARM method exclusively studies authentication in ubiquitous computing. In reviewing current existing researches, the researcher believes that several researches regarding security have been accomplished and published. At the time of research, the researcher discovered that ubiquitous computing must have mobility, entity authentication, corresponding entity authentication, outgoing data authentication, and connection/non-connection confidentiality as basic requirements. Thus, this paper proposes the adoption of PMI to meet the requirements listed above and to implement the authentication for the smart device. The ubiquitous computing devices lacked computing, storing, and other capabilities. But since a device must meet the requirements discussed earlier, applying PMI on top of the currently used encryption system will satisfy the device capabilities and requirements. Since all devices can carry out the authentication and access control with the PMI certificate, only activities authorized to the devices will be allowed. I will also propose a method that uses a PMI certificate for a device. The general system flow will be discussed after the consideration for the proposed method is reviewed.

4.1 Consideration for Proposed Scheme

The goal of the proposed method is to provide a device retaining the user information of the previous domain even when the device is moved into the multi domain. Thus, the following must be considered for the proposed method.

- A user device alone can be moved and this user device can be linked to other devices: This means that when a user's smart device is moved into the multi domain, the smart device can be linked to devices in the multi domain to receive services. User information must therefore be extracted from the smart device, as other devices exist for services only.

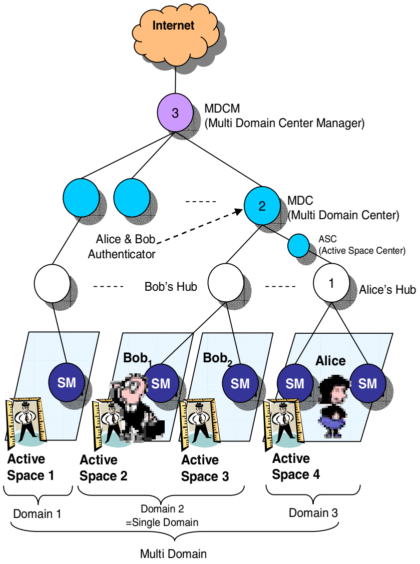


Fig. 1. Whole Flow of Proposed Scheme

A user device is authenticated through the Hub where all device information is stored in the same space. This device is authenticated through the MDC (Multi Domain Center) when moved to a different space: When a user device is located in the user domain, the user device can be authenticated through the Hub, which connects all user spaces. When the user device moves away from the user space, the smart device can also be authenticated through the user's Hub. But when a smart device is moved, an authentication method other than the Hub must be used. That method is to use the MDC, which authenticates the smart device in the multi domain.

- Initial authentication information is granted to a smart device through a user's hub from MDC: During this process, an authorized user registers devices in the user's hub. If a user creates authentication information from the MDC for the first time, the authentication information is stored in the hub and to the smart device. At this point, higher-level MDC authenticates the smart device using the created authentication information given to the user.

- At this step, the privacy of the user location is not considered..

Figure 1 shows the entire flow of the system. User registration and device registration must be done in advance in the single domain and the multi domain. These operations must be done in order for the initial MDC to grant a PMI certificate to a user. The next operation is to authenticate in the single and the multi domain. The authentication in the single domain is shown below. Let us assume that Bob has spaces where he can move around. Let us also assume that there are two active spaces for Bob: Bob1 and Bob2, and these two active spaces are located in Domain_2 (a single domain). When Bob's Device_B moves away from Active Space 2 to Active Space 3, the single domain authentication occurs. Bob's SM_B in Active Space 2 notifies his Hub of movement, transmits the authentication information to Device_B and the Hub, and requests the authentication from Active Space 3. At this point, Active Space 2 sends the SM_B authentication information to Active Space 3 for efficient authentication. Active Space 3 receives the request and compares the authentication information from Active Space 2 and the one from the Hub, and then authenticates SM_B in Active Space 3. All these steps complete the single domain authentication. In the multi-domain authentication, when user Bob moves to Alice's Domain_A, he can use Alice's Device_A after getting SM_B authenticated. In this case, Bob's SM_B sends the movement signal to his Hub and requests the authentication after the move to Alice's Active Space. At this point, the authentication information is requested through Alice's Hub. Bob requests the authentication from MDC through Alice's Hub. If Bob's information does not exist in MDC, the authentication request is made against MDCM, which is a higher entity. In this way, although an entity is not systematically contained in or connected to the higher entity, the authentication using devices in other spaces can be completed through the Internet.

4.2 System Parameters

Next, system parameters used in this method are explained. Each parameter is distinguished according to its components. The components create and transmit the parameters.

*: (SM: SMart device, D: Device, SD: Single Domain, MDC: Multi Domain Center, A: Alice, B: Bob, MDCM: MDC Manager, ASC: Active Space Center)

$Cert\ *$: public key of * including Certification

$PCert\ *$: public key of * including PMI Certification

n : PMI Certification maximum issues number

AP : Available Period

i : user issued device

pw : password

$ID\ *$: * of Identity

$Hub\ *$: * of Hub

r : user Hub generated random number

$E_*(\ )$: * key with Encryption

$R*$: * of authority

$H(\ )$: Secure Hash Function

4.3 Proposed Scheme

The detailed flow of these proposed methods is described below. In the first method, when a user moves to his domain with his smart device and attempts to use devices in the new domain, the user is authenticated using the smart device in which the user authentication information is stored. In the second method, when a user moves to the multi domain and attempts to use devices there, the user is authenticated using the smart device in which his user authentication information is stored.

4.3.1 User Registration and Device Registration

A user must have authentication information for all devices in the initial stage to use the devices in the single domain. A user receives a certificate from the MDC (Multi Domain Center), and his devices are granted a PMI (Privilege Management Infrastructure) certificate through a Hub. PMI certificates are granted according to the mutually agreed methods with the MDC. Granted PMI certificates are stored in a smart device.

Step 1. The following processes are required to create Device_A authentication for User A. The MDC grants User A a certificate which allows User A to create n number of PMI certificates. A PMI certificate consists of the User A ID, privilege, and effective period of the certificate.

$$MDC \rightarrow Hub_A : Cert_A[ID_A, R_A, n, AP]$$

Step 2. User A grants PMI certificates to Device_A and SM_A using the granted certificate. The certificate contains the path to a higher-level certificate.

$$Hub_A \rightarrow SD_A(orDevice_A) : PC_A = PCert_A[ID_A, H(Cert_A \| r), i] \| AP$$

$$SD_A : E_{PK_{DDC}}[PC_A]$$

$$SD_A\, install : E_{pw(orPIN)} \lfloor E_{PK_{DDC}}[PC_A] \rfloor$$

Step 3. User A informs the MDC of the certificate granted to his Device_A. Afterwards, User A's PMI certificate is used, and User A is authenticated using the PMI certificate path within SM_A.

$$Hub_A \rightarrow MDC : E_{PK_{DDC}}[H(Cert_A \| r), r, i]$$

4.3.2 Authentication in the Single Domain

When SM_A on User A's Hub attempts to use Device_A2 in AS_A2, SM_A uses existing information as is.

Step 1. SM_A exists in the active space AS_A1 and sends the movement signal to Device_A1 when the movement occurs.

$$SD_{A_1} \rightarrow Device_{A_1} : Signal(Outgoing)$$

Step 2. Device_A1 notifies Hub A of SM_A movement.

$$Device_{A_1} \rightarrow Hub_A : E_{PK_{Hub}} \lfloor Device_{A_2}, E_{PK_{hub}}(PC) \rfloor$$

Step 3. Device_A1 also transmits SM_A information to Device_A2.

$$Device_{A_1} \rightarrow Device_{A_2} : \lfloor Device_{A_1} \| E_{pw(orPIN)} E_{PK_{Hub}}(PC) \rfloor$$

Step 4. Device_A2 uses the authentication information received from Device_A1 to send its information to Hub A.

$$Device_{A_2} \rightarrow Hub_A : \lfloor Device_{A_2} \| E_{pw(orPIN)} E_{PK_{Hub}}(PC) \rfloor$$

Step 5. Hub A also confirms the authentication information received from Device_A1 by comparing it to the information received from Device_A2, and then approving the SM_A authentication.

$$Hub_A : E_{pq\,(orPIN)}E_{PK_{Hub}}(PC) = E_{PK_{Hub}}(PC)$$

$$Compare : (Device_{A_1})E_{PK_{Hub}}(PC) \overset{?}{=} (Device_{A_1})E_{PK_{Hub}}(PC)$$

Step 6. Hub A completes the confirmation and accepts the authentication for SM_A.

$$Hub_A \rightarrow Device_{A_2} : [Device_{A_2} \| Auth_{SD}]$$

Step 7. After SM_A provides its values and compares the values, it approves the use of Device_A2 in the active space.

4.3.3 Authentication in the Multi domain

When SM_A in Domain_A moves to Domain_B and uses User A's information to use Domain_B and Device_B, SM_A uses User A's information as is.

Step 1. A movement signal is sent using Device_A in Domain_A. If Hub A receives the movement signal from SM_A, it removes itself from the space list.

$$SD_{A_1} \rightarrow Hub_A : Signal\,(Outgoing)$$

$$Hub_A : SD_{DeviceList} \rightarrow Delete\,[SD_{A_1}]$$

Step 2. Hub A notifies the MDC that it is moving out of Domain_A. If it moves to a different MDC, it notifies MDCM.

$$Hub_A \rightarrow MDC : (ID_A, i)$$

$$MDC \rightarrow MDCM : (ID_A, i)$$

Step 3. After notification that SM_A is finally located in Domain_B, it requests authentication from Device_B in Domain_B.

$$SD_{A_1} \rightarrow Hub_B : Signal\,(Ongoing)$$

$$SD_{A_1} : E_{pw}[E_{PK_{DDC}}[PC_A]] = E_{PK_{DDC}}[PC_A]$$

$$SK_{A_1} \rightarrow Device_B : E_{PK_{DDC}}[PC_A]$$

$$Device_B \rightarrow Hub_B : E_{PK_{Hub_B}}[PC_B, E_{PK_{DDC}}[PC_A]]$$

Step 4. Hub B in Domain_B verifies the authentication information from Device_B.

$$Hub_B : E_{SK_{Hub}}[E_{PK_{Hub_B}}[PC_B, E_{PK_{DDC}}[PC_A]]] = PC_B, E_{PK_{DDC}}[PC_A]$$

$$Hub_B : PC'_B \overset{?}{=} PC_B$$

Step 5. If the Device_B authentication information is passed, Hub B transmits the authentication information to the MDC.

$$Hub_B \rightarrow MDC : (ID_B, E_{PK_{MDC}}[E_{PK_{DDC}}[PC_A] \| ID_B])$$

Step 6. The MDC verifies that the authentication information is generated from Domain_B User. If confirmed, the MDC approves the authentication for SM_A.

$$MDC : E_{PK_{DDC}}[PC_A] \| ID_B$$

$$MDC : PC'_A = PCert_A[ID_A, H(Cert_A \| r), i]$$

Step 7. In Domain_B, Hub 2 accepts the received authentication for SM_A, and allows for the use of Device_B in Domain_B.

5 Comparison with Proposed Scheme and JARM Scheme

This chapter will attempt to analyze the proposed protocol by classifying the user and device registration and the authentication in the multi domain, and compare the protocol to the existing method. In the existing method, SM_A is not authenticated by moving to AS1. Therefore, the research in this paper seeks to put emphasis on how to authenticate a user who wishes to use Device_B by using user information A when SM_A is moved to Domain_B. The existing methods attempt to solve the problem by assigning different authentication information to the devices. But the weak point of this approach is to require all devices to be available when the entire authentication information is obtained. This method raises a problem in that authentication information cannot be obtained if a device is lost.

The proposed method and the existing method are compared below. The details of the proposed method will also be discussed below.

- Mobility: A device containing authentication information that a user owns can use all services. In the proposed method, a PMI certificate is included in the device. Thus, the device can be moved away from the other devices, and can also be linked to a different device.
- Entity Authentication: Although a smart device is moved away from its domain, the device in the multi domain can be authenticated using the previous user's information. Mobility is guaranteed because a user device uses its PMI certificate. However, if a certificate is not protected, it can be used by malicious users who can also be authenticated. To resolve this issue, the proposed method protects the PMI certificate by using device access and certificate access protection mechanisms such as the password and PIN.
- Corresponding Entity Authentication: When a device is located in Domain_A, corresponding entity authentication is provided to verify that Device_B and B are identical entities. This authentication method implements device authentication through the entity of the previous user when multiple devices are connected to one domain. This authentication can provide different levels of protection. Even when a smart device is moved, the authentication can be done using what is stored in the smart device. Also, a PMI certificate, which is an internal certificate and identical to the certificate from User A, can be used when performing the corresponding entity authentication.
- Connection/Non-connection Confidentiality: Device_B in Domain_A must provide confidentiality for user data in both Domain_A and Domain_B. Domain_A receives information from B and receives the final authentication from the higher level. Non-connection confidentiality must provide data confidentiality before Device_B connects to a specific domain.

6 Conclusion

Rapid expansion of the Internet has required a ubiquitous computing environment that can be accessed anytime anywhere. In this ubiquitous environment, a user ought to be given the same service regardless of connection type even though the user may not specify what he needs. Authenticated devices that connect user devices must be used regardless of location. If a device is moved to another user space from a previous user space, the authentication must be performed well in the transferred space. This is so because a device is not restricted to the previous authentication information, but can use new authentication information in the new space. This paper attempts to solve the problems discussed earlier by utilizing such entities as the Hub, ASC, and MDC in order to issue PMI certificates to devices that do not have computing capability. This provides higher-level devices the authentication information of the smart devices in order to authenticate the movement of these smart devices. With this proposed method, if a smart device requests the authentication after moving to the multi domain, the authentication is performed against the devices in the domain where the smart device belongs, and the smart device requests the authentication from the MDC. In the user domain, the authentication is performed through the Hub. However, the authentication is performed through MDC when a device is moved to the multi domain environment. This proposed method, therefore, attempts to solve the existing authentication problem. With regard to the topics of privacy protection, which is revealed due user movement key simplification (i.e., research on a key that can be used for a wide range of services), and the provision of smooth service for data requiring higher bandwidth, the researcher has reserved them for future researches.

Reference

[1] A. Aresenault, S. Tuner, Internet X.509 Public Key Infrastructure, Internet Draft, 2000.

[2] ITU-T, Draft ITU-T RECOMMANDATION X.509 version4, ITU-T Publications, 2001.

[3] Jalal Al-Muhtadi, Anand Ranganathan, Roy Campbell, and M. Dennis Mickunas,"A Flexible, Privacy-Preserving Authentication Framework for Ubiquitous Computing Environments," ICDCSW '02, pp.771-776, 2002

[4] Mark Weiser,"Hot Topics: Ubiquitous Computing," IEEE Computer, October 1993

[5] M. Roman, and R. Campbell, "GAIA: Enabling Active Spaces," 9th ACM SIGOPS European Workshop, September 17th-20th, 2000, Kolding, Denmark

[6] S. Farrell, R. Housley, An Internet Attribute Certificate Profile for Authorization, Internet Draft, 2001.

[7] Sanjay E. Sarma, Stephen A. Weis and Saniel W. Daniel , White Paper:RFID Systems, Security & Privacy Implications, AUTO-ID Center, MIT, Nov, 2002

[8] Gen-Ho, Lee, "Information Security for Ubiquitous Computing Environment", Symposium on Information Security 2003, KOREA, pp 629-651, 2003

[9] Sung-Yong Lee and Hyun-Su Jung, "Ubiquitous Research Trend & Future Works", Wolrdwide IT Vol. 3, No. 7, pp 1-12, 2002

[10] Yun-Chol Lee, "Home Networks Technology & Market Trend", ITFIND Weeks Technology Trend(TIS-03-20) No. 1098, pp22-33, 2003

[11] Aura Project home page. http://www-2.cs.cmu.edu/~aura/

[12] CoolTown home page. http://www.cooltown.hp.com

[13] IBM Websphere home page.http://www-3.ibm.com/software/info1/websphere/index.jsp?tab=highlights

[14] Microsoft Research, EasyLiving Website, http://www.research,microsoft.com/easyliving

[15] Oxygen Project home page. http://oxygen.lcs.mit.edu/

[16] Portolano home page. http://portolano.cs.washington.edu/

[17] TRON Project home page. http://www.tron.org/index-e.html

Improving the CGA-OMIPv6 Protocol for Low-Power Mobile Nodes

Ilsun You

Department of Information Science, Korean Bible University,
205 Sanggye-7 Dong, Nowon-ku, Seoul, 139-791, South Korea
isyou@bible.ac.kr

Abstract. In order to enhance the route optimization mode, the OMIPv6 and CGA-OMIPv6 protocols have been proposed. They use public key methods to minimize the amount of signaling messages and handover latency. However, since the public key operations are computationally expensive, it is so difficult for the protocols to support low-power mobile nodes. In this paper, we propose an enhanced OMIPv6 protocol. By employing the home agent as a security proxy, the proposed protocol improves the CGA-OMIPv6 protocol to support low-power mobile nodes. As a result, the proposed protocol significantly reduces the computational cost of the mobile node into just two HMAC and two hash operations. Furthermore, it achieves good scalability and manageability through the bind between the home agent's address and public key.

1 Introduction

Mobile IP Version 6 (MIPv6) provides the route optimization (RO) mode to allow the mobile node (MN) and its correspondent node (CN) to exchange packets directly [1]. In the mode, the MN performs the binding update (BU) process to continually inform its home agent (HA) and CN about its new care-of-address (CoA). However, if the BU process is not authenticated, the mode may expose the involved parties to various security threats. In order to secure the process, the mode employs the return routability (RR) procedure, where the CN verifies the MN's home address (HoA) and CoA, and exchanges a shared secret with the MN.

Although the mode is efficient, it results in the following problems [1-3]:

- Because of security reasons, the mode restricts the lifetime of the shared secret to maximum 420 seconds. Such a limited lifetime forces the CN and the MN to update their shared secret at a high frequency, thus causing the number of mobility signaling messages and handover latency to be increased.
- The RR procedure's messages are exchanged in clear in the MN-CN path and the HA-CN path across the Internet. This makes the mode vulnerable to various security threats every few minutes during the ongoing session.

In order to solve the above problems, the Optimized MIPv6 (OMIPv6) protocol and its successor, the Optimized Mobile IPv6 with CGA (CGA-OMIPv6) protocol, have been proposed [2,3]. They use public key methods to compute a strong shared

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 336 – 343, 2006.

secret, the lifetime of which is sufficient long to minimize the amount of signaling messages and handover latency. However, since the public key operations are computationally expensive, it is so difficult for the protocols to support wireless low-power MNs that cannot afford to provide sufficient computing power for public key cryptography. Furthermore, the CGA-OMIPv6 protocol suffers from the lack of scalability and the high management cost because in the protocol each MN's HoA is generated from its public key and used as a Cryptographically Generated Addresses (CGA) [4].

In this paper, we improve the CGA-OMIPv6 protocol to support low-power MNs and address the scalability and manageability problem. For that, we employs the HA as a security proxy, which performs all the expensive operations on behalf of its MNs.

The rest of the paper is organized as follows. Section 2 reviews the CGA-OMIPv6 protocol and analyzes its drawbacks. In section 3, we propose an enhanced OMIPv6 protocol. Section 4 analyzes the proposed protocol. Finally, section 5 draws some conclusions.

2 Review of CGA-OMIPv6 Protocol

The CGA-OMIPv6 protocol applies CGAs to improve the efficiency and security of the MIPv6 route optimization [3]. Such an enhanced route optimization can reduce the number of mobility signaling messages and handover latency while offering strong security. This section briefly reviews the protocol and analyzes its drawbacks.

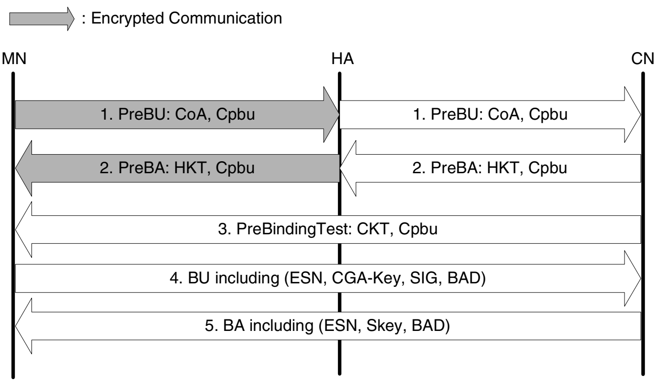
2.1 Notation

- | denotes concatenation.
- PrK_X denotes X's private key.
- PuK_X denotes X's public key.
- ESN denotes Extended Sequence Number.
- MH-Data1 denotes the content of the Mobility Header where the authenticator in the Binding Authorization Data option is zeroed.
- MH-Data2 denotes the content of the Mobility Header excluding the Authenticator in the Binding Authorization Data option.
- CGA-Key denotes the CGA public key and other parameters.
- $P_X(m)$ means that the message m is encrypted with the public key of X.
- $P_X{}^{-1}(m)$ means that the message m is encrypted with the private key of X.
- *p* and *g* denote the public Diffie-Hellman parameters. *p* is a large prime and *g* is a generator of the multiplicative group Zp*. It is assumed that they are agreed upon previously by all nodes participating in the Diffie-Hellman key exchange.

2.2 Protocol Operation

The CGA-OMIPv6 protocol is divided into two phases: the initial phase and the subsequent movement phase. The initial phase verifies the MN's CGA ownership and then makes the MN and the CN exchange the semi-permanent security association key, Kbmperm. During the subsequent movement phase, the MN and the CN efficiently and securely authenticate BU and BA messages through the exchanged Kbmperm.

2.2.1 Initial Phase


: Encrypted Communication
MN
HA
CN
1. PreBU: CoA, Cpbu
1. PreBU: CoA, Cpbu
2. PreBA: HKT, Cpbu
2. PreBA: HKT, Cpbu
3. PreBindingTest: CKT, Cpbu
4. BU including (ESN, CGA-Key, SIG, BAD)
5. BA including (ESN, Skey, BAD)

Cpbu = prebinding update cookie
K_{CN} = CN's secret key
HKT = First(64, HMAC_SHA1(Kcn, HoA|nonce|0))
CKT = First(64, HMAC_SHA1(Kcn, CoA|nonce|1))
SIG = P^{-1}_{MN}(SHA1(CN|CoA|MH-Data1))
Kbmperm = a secret key randomly generated by CN
SKey = P_{MN}(Kbmperm)
Kbm = SHA1(HKT|CKT)
BAD = HMAC_SHA1(Kbm, CN|CoA|MH-Data2)

Fig. 1. Initial Phase of the CGA-OMIPv6 protocol

The initial phase is described in Fig. 1. In steps 1-3, this phase performs both HoA and CoA tests to verify if the MN's HoA and CoA are reachable. Also, it allows the MN and the CN to use the keygen tokens provided in steps 2 and 3 to establish a shared secret key, Kbm, which is then used to prevent denial of service attacks in steps 4 and 5. In step 4, upon receiving the BU message, the CN first checks BAD with Kbm before performing the asymmetric cryptographic operation. If BAD is valid, the CN can ensure that the MN receiving messages of steps 2 and 3 sent the BU message. In this case, the CN verifies the MN's CGA public key and signature while defending against denial of service attacks. In step 5, the CN generates Kbmperm as the semi-permanent security association key and encrypts it with the MN's public key. Then, the CN sends the MN the encrypted Kbmperm in the SKey option of the BA message. When receiving the BA message, the MN verifies BAD and then decrypts the encrypted key with its own private key.

2.2.2 Subsequent Movement Phase

As mentioned above, the initial phase performs the HoA test, verifies the MN's CGA ownership and then establishes Kbmperm with the help of the CGA public key. This enables the CN to have strong assurance about correctness of the MN's HoA. Thus, the subsequent movement phase does not need to perform the HoA test again. Fig. 2 shows this phase, where the standard binding update procedure is optimized by doing only the CoA test. Also, the standard procedure's security is enhanced by using Kbmperm' instead of the normal Kbm. As a result, this phase is able to significantly reduce both the signaling and handover latency while achieving strong security.

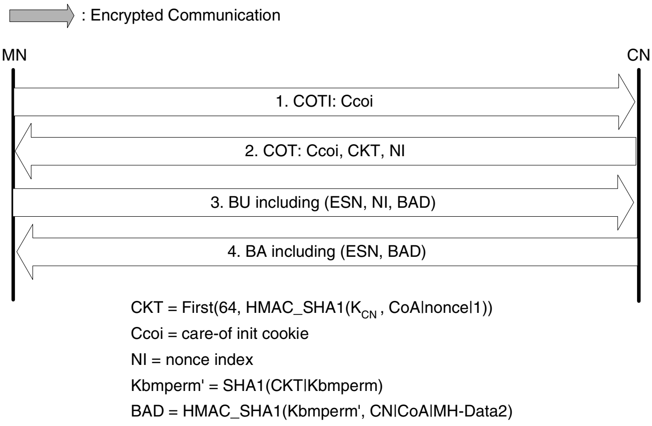

: Encrypted Communication
MN
CN
1. COTI: Ccoi
2. COT: Ccoi, CKT, NI
3. BU including (ESN, NI, BAD)
4. BA including (ESN, BAD)
CKT = First(64, HMAC_SHA1(K_CN , CoA|nonce|1))
Ccoi = care-of init cookie
NI = nonce index
Kbmperm' = SHA1(CKT|Kbmperm)
BAD = HMAC_SHA1(Kbmperm', CN|CoA|MH-Data2)

Fig. 2. Subsequent Movement Phase of the CGA-OMIPv6 protocol

2.3 Drawbacks of the CGA-OMIPv6 Protocol

The drawbacks of the CGA-OMIPv6 protocol are as follows:

- Since the CGA-OMIPv6 protocol depends on the CGA method, it includes computationally expensive cryptographic operations in the initial phase. That is, as analyzed in Table 1, the MN should perform one sign operation and one public key decryption while the CN should perform one verify operation and one public key encryption. This may be a critical limitation to low-power mobile nodes that cannot afford to provide sufficient computing power for public key cryptography. Thus, the protocol needs to be enhanced for such nodes.
- In this protocol, each MN's HoA is generated from its public key and used as a CGA. However, such a bind is not desirable for the following reasons [5]. First, the MN's HoAs tend to be reassigned when network configurations change or when service providers change, thus not being as persistent like domain names and email-addresses. This may be an obstacle to applying the protocol because the CGA method increases the cost of address generation by a factor of 2^{16*Sec} as well as the cost of brute-force attacks [4]. Second, the MN's HoA cannot be used as CGA in a network environment where automated dynamic address assignment is employed.

Table 1. Computational Cost of the Initial Phase

Step	MN	CN
1	0	0
2	0	HKT: 1×HMAC_SHA1
3	0	CKT: 1×HMAC_SHA1
4	Kbm: 1×SHA1 SIG: 1×SIGN BAD: 1×HMAC_SHA1	HKT: 1×HMAC_SHA1 CKT: 1×HMAC_SHA1 Kbm: 1×SHA1 BAD: 1×HMAC_SHA1 CGA: 2×SHA1 SIG: 1× VERIFY
5	BAD: 1×HMAC_SHA1 Skey: 1×PK_DEC	SKey: 1×PK_ENC BAD: 1×HMAC_SHA1

* PK_ENC denotes the cost for a public key encryption
* PK_DEC denotes the cost for a public key decryption
* SIGN denotes the cost for generating a digital signature
* VERIFY denotes the cost for verifying a digital signature

3 Proposed Protocol

In this section, we propose an advanced OMIPv6 protocol, which improves the CGA-OMIPv6 protocol for low-power mobile nodes. Because only the initial phase includes asymmetric cryptographic operations, we focus on and enhance the phase. Thus, the subsequent movement phase of our protocol is as same as one of the CGA-OMIPv6 protocol.

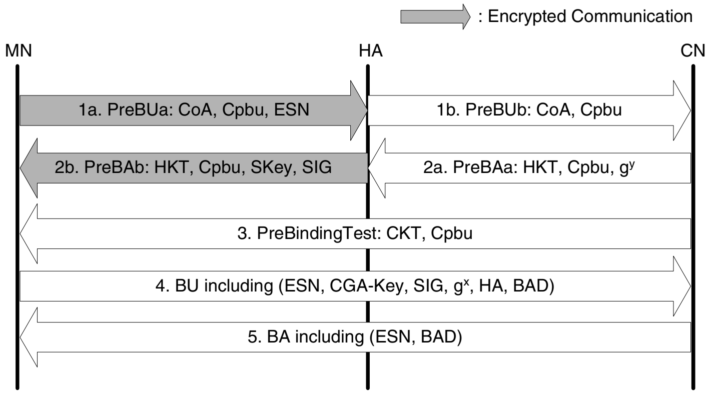
3.1 Security Proxy

To support low-power mobile nodes, our protocol employs the HA as a security proxy, which performs all the expensive operations on behalf of the MN. For that, in our protocol, the HA has the public/private key pair PuK_{HA}/PrK_{HA} and uses its own IPv6 address derived from PuK_{HA} as a CGA. Therefore, as a security proxy for the MN, it can generate a signature, SIG, while exchanging Kbmperm with the CN. Such a mechanism is available in the MIPv6, because communication between the MN and the HA is protected with the pre-establish security association [1,5-7].

3.2 Protocol Operation

The proposed protocol is shown in Fig. 3. Unlike the CGA-OMIPv6 protocol, this protocol uses the Diffie-Hellman method to allow Kbmperm to be exchanged during the step 2.

Step 1 (PreBUa and PreBUb): The MN sends the PreBUa message to the CN to start the initial phase. When the message arrives at the home link, it is intercepted by the HA using IPv6 Neighbor Discovery [1,8]. The HA keeps ESN and Cpbu in this message. ESN is used to generate a signature, SIG, for the PreBAb message and Cpbu is used to protect the HA against denial of service attacks.


: Encrypted Communication
MN
HA
CN
1a. PreBUa: CoA, Cpbu, ESN
1b. PreBUb: CoA, Cpbu
2b. PreBAb: HKT, Cpbu, SKey, SIG
2a. PreBAa: HKT, Cpbu, g^y
3. PreBindingTest: CKT, Cpbu
4. BU including (ESN, CGA-Key, SIG, g^x, HA, BAD)
5. BA including (ESN, BAD)

Cpbu = Prebinding update cookie
K_{CN} = a secret key of CN
HKT = First(64, HMAC_SHA1(K_{CN}, HoA|nonce|0))
CKT = First(64, HMAC_SHA1(K_{CN}, CoA|nonce|1))
SIG = P^{-1}_{HA}(SHA1(CN|CoA|ESN|g^x|CGA-Key|HKT))
SKey = SHA1(g^{xy}|HKT)
Kbm = SHA1(HKT|CKT)
Kbmperm = SHA1(SKey|CKT)
BAD = HMAC_SHA1(Kbm, CN|CoA|MH-Data2)

Fig. 3. Initial Phase of the Proposed protocol

Steps 2-3 (PreBAa, PreBAb and PreBindingTest): Upon receipt of the PreBUb message, the CN sends the PreBAa message including its Diffie-Hellman public key, g^y, to the MN's HoA, while sending the PreBindingTest message to the MN's CoA. When the HA intercepts the PreBAa message using IPv6 Neighbor Discovery, it first validates Cpbu to prevent denial of service attacks. If the value is valid, it computes SKey = h(g^{xy} | HKT), generates a signature, SIG, with PrK_{HA} and sends the PreBAb message to the MN. Thus, in step 2, the HA off-loads the asymmetric cryptographic operations of the MN to itself. To prevent denial of service attacks, the HA and the CN use the same g^x and g^y as their Diffie-Hellman public keys for each protocol run instead of generating new values. After receiving the PreBAb and PreBindingTest messages, the MN computes Kbm and Kbmperm for the next step.

Steps 4-5 (BU and BA): Unlike the CGA-OMIPv6 protocol, the BU message includes the HA's CGA-key, the HA's Diffie-Hellman public key, g^x, and the HA's address HA, which are used to verify HA's CGA and negotiate Kbmperm. If the BU message is valid, the CN can ensure that the MN's CoA and HoA are reachable and the Diffie-Hellman public key, g^x, is valid. In this case, it computes Kbmperm for the following phase, while sending the BA message to the MN.

4 Analysis

Table 2. Computational Cost of the Proposed Protocol

Step	MN	HA	CN
1	0	0	0
2	0	Skey: 1×DH_ AGREE + 1×SHA1 SIG: 1×SIGN	HKT: 1×HMAC_SHA1
3	0	0	CKT: 1×HMAC_SHA1
4	Kbm: 1×SHA1 BAD: 1×HMAC_SHA1 Kbmperm: 1×SHA1	0	HKT: 1×HMAC_SHA1 CKT: 1×HMAC_SHA1 Kbm: 1×SHA1 BAD: 1×HMAC_SHA1 CGA: 2×SHA1 SIG: 1×VERIFY SKey: 1×DH_ AGREE + 1×SHA1
5	BAD: 1×HMAC_SHA1		BAD: 1×HMAC_SHA1

* DH-AGREE denotes the cost for a Diffie Hellman key exchange

Table 3. Computational Cost Comparison of the Proposed Protocol and the CGA-OMIPv6 Protocol

Step	MN	HA	CN
Proposed Protocol	2×SHA1 2×HMAC_SHA1	1×DH_ AGREE 1×SIGN 1×SHA1	6×HMAC_SHA1, 4×SHA 1×VERIFY 1×DH_ AGREE
CGA-OMIPv6 Protocol	1×SHA1, 2×HMAC_SHA1 1×PK_DEC 1×SIGN	0	6×HMAC_SHA1, 3×SHA1 1×VERIFY 1×PK_ENC
Difference	+ 1×SHA1 - 1×PK_DEC - 1× SIGN	+ 1×DH_ AGREE + 1×SIGN + 1×SHA1	+ 1×SHA + 1×DH_ AGREE - 1×PK_ENC

Computational Cost: Table 2 analyzes the computational cost of the proposed protocol, which is then compared with that of the CGA-OMIPv6 protocol in Table 3. As shown in the tables, the proposed protocol significantly reduces the computational cost of the MN from (1×SHA1 + 2×HMAC_SHA1 + 1×PK_DEC + 1×SIGN) to 2×(SHA1+ HMAC_SHA1). The security proxy HA does the expensive cryptographic operations (1×DH_ AGREE + 1×SIGN) on behalf of the MN so the MN does not need to perform public key operations. The reduced cost makes it possible for the protocol to support low-power mobile nodes.

Public Key Mechanism: In the proposed protocol, the HA's address is derived from its public key and used as a CGA. Such a mechanism enables the protocol to overcome the drawbacks presented in section 2.2 because of the following reasons. First, the HA's address is normally much more persistent than the MN's HoA. Second, the number of HAs is significantly smaller than that of MNs. This can give the good scalability and reduce the cost for managing CGAs. Such reduced cost allows the proposed protocol to generate CGAs more securely through high Sec values such as 5, 6 and 7.

5 Conclusions

In this paper, we proposed an enhanced OMIPv6 protocol, which improves the CGA-OMIPv6 protocol to support low-power MNs and address the scalability and manageability problem. The proposed protocol employs the HA as a security proxy that performs all the expensive operations on behalf of the MN. Also, the HA has the public/private key pair and uses its own IPv6 address derived its public key as a CGA. As a result, the proposed protocol significantly reduces the computational cost of the MN from two asymmetric cryptographic, two HMAC and one hash operations to two HMAC and two hash operations. Furthermore, through the bind between the HA's address and public key, it reduces the cost for managing CGAs while achieving good scalability and manageability.

References

1. D. Johnson, C. Perkins and J. Arkko, "Mobility Support in IPv6," RFC 3775, June 2004
2. W. Haddad, F. Dupont, L. Madour, S. Krishnan and S. Park, "Optimizing Mobile IPv6 (OMIPv6)," draft-haddad-mipv6-omipv6-01.txt, Feb. 2004 (Work in progress)
3. W. Haddad, L. Madour, J. Arkko and F. Dupont. "Applying Cryptographically Generated Addresses to Optimize MIPv6 (CGA-OMIPv6)," draft-haddad-mip6-cga-omipv6-04, May 2005 (Work in progress)
4. T. Aura, "Cryptographically Generated Addresses (CGA)," RFC 3972, March 2005
5. R. Deng, J. Zhou, and F. Bao, "Defending Against Redirect attacks in Mobile IP," Proceedings of the 9th ACM Conference on Computer and Communications Security, Nov. 2002
6. Y. Lee, I. You and S. Rhee, "Improving CAM-DH Protocol for Mobile Nodes with Constraint Computational Power," KES 2004, Springer-Verlag LNCS 3215, pp. 67-73, Sept. 2004
7. I. You and K. Cho, "A Security Proxy Based Protocol for Authenticating the Mobile IPv6 Binding Updates," ICCSA 2004, Springer-Verlag LNCS 3043, pp. 167-174, May 2004
8. T. Narten, E. Nordmark, and W. Simpson, "Neighbor Discovery for IP Version 6 (IPv6)," RFC 2461, Dec. 1998.

Tracking Illegal System Access in a Ubiquitous Environment – Proposal for ATS, a Traceback System Using STOP

Gwanghoon Kim[1], Soyeon Hwang[1], and Deokgyu Lee[2]

[1] INFOSEC, Co.LTD., 7F, Saemaul-undong B/D 1008-4, Daechi-dong, Kangnam-gu,
Post Code 135-851, Seoul, Korea
{Jeric, so}@skinfosec.com
[2] Divison of Information Technology Engineering, SoonChunHyang University,
Eupnae-ri, Shinchang-myun, Asan-si, ChoongNam, Korea
hbrhcdbr@sch.ac.kr

Abstract. In a ubiquitous environment, the boundaries of network topology can change irregularly. In this paper, an Agent-based Traceback System (ATS) is proposed to track attacks that utilize systems within an area of the network topology that has been marked for management purposes. Some of the information exchanged within the proposed system utilizes the previously verified STOP [1]. The additional information provided by utilizing the ATS proposed in this paper will greatly enhance the reliability of the traceback process. In addition, the proposed system is flexible enough to be applied to resource management systems as well.

1 Introduction

When a ubiquitous concept is applied to the existing LAN environment, previous limitations that restricted the roles of each network component no longer become valid. In addition, the boundaries that once separated one network from another lose meaning in a ubiquitous environment. The systems that comprise a network can take on more than one network interface (both fixed and wireless), enabling simultaneous access from both inside and outside the network. This means that the entire topology of the network can be altered, and a system within the network can be used by an intruder as an access path.

In this paper, an Agent-based Traceback System (ATS) with the following characteristics is proposed. A light Traceback Agent Module (TAM) is installed on all systems connecting the managed area to the outside area. The TAM attaches additional information to the STOP, which has been previously proposed by Brian Carrier and Clay Shields as a means of providing traceback information on processes executed by systems on the connection path. Additional information generated by TAM is attached to the basic traceback information provided by STOP. This information is then sent to a Traceback Server (TS), which is installed in the network to collect information on outside access. When an intrusion occurs, information stored in the Traceback Server is used to traceback the path of the intrusion.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 344 – 351, 2006.

This paper is organized as follows. In section 2, research trends of traceback technology is examined, including STOP, which is utilized in the proposed ATS. In Section 3, the components and principles of the proposed ATS are explained, and a hypothetical situation is considered in order to illustrate the execution procedure of the proposed system. A conclusion and discussions for future research direction are presented in Section 4.

2 Related Work

2.1 Research Trends of Traceback Methods

Currently, there are two major categories of studies taking place regarding Network Traceback. The first is IP traceback, which is researched in order to understand the source of a spoofed packet. The main purpose of IP traceback is to find the source of the packets used in a DoS attack. The second is the study of using the services of systems that have been utilized in the intrusion for traceback.

- ***IP Traceback:*** Proactive Traceback is a method of generating and attaching/ transmitting traceback information during the packet transfer process. Thus, if an intrusion occurs, the traceback information that has already been sent is compiled in order to trace the source of the attack. Proactive Traceback methods include Packet Marking [2][3] and ICMP traceback message-based traceback method [4].

  Reactive Traceback is a method of tracing the origin of an attack when the connection established by the intrusion is still active. Some examples of Reactive Traceback are hop-by-hop Traceback [5][6], Hash based IP Traceback [7], and Traceback using IPSec technology [8].

- ***Protocol-Based Connection-Chain Traceback:*** The only research results that have been published regarding the Protocol-Based solution is Caller Identification System (Caller ID), which was introduced in 1993 [9].

  In this system, mutual authentication between the client and server is necessary in order to connect to the system, and an Extended TCP Wrapper and Caller Identification Server (CIS) must be established in the two connecting systems.

- ***The Identification Protocol*:** IDENT was originally called the Authentication Server Protocol [10], but was changed to the present form to reflect its activation method. IDENT is a simple two-way protocol that allows the target server to know the identity of the client attempting the connection.[11]

  IDENT is basically provided in UNIX or LINUX as a daemon, and is used in some applications like IRC and Sendmail.

2.2 The Session Token Protocol (STOP) [1]

Proposed by Brian Carrier and Clay Shields, this protocol attaches additional functions to the IDENT used in most UNIX systems. The Session Token Protocol (STOP) was developed with the following design goals.

First, STOP must be compatible with the IDENT specified in RFC 1413, as IDENT is already in widespread use. Also, the daemon which implements this protocol should have a mechanism that enables the saving of both user-level and application–level data, and STOP should also include a mechanism that enables the tracing of a client's path through previous hosts. Finally, when a request is received, the response to the request should not be delayed, and a considerable load should not be added to the daemon.

In this study, the Session Token Protocol proposed by Brian Carrier and Clay Shields is used to achieve communication between the TAM and TS.

When an intrusion is detected within the internal network, the TS requests traceback information from the system where the intrusion is being attempted. The request message takes on the following format:

< CL_PORT >,< SV_PORT >:< REQ_TYPE >[:< SID >][:< CL_IP >]

Where, CL_PORT is the port of the Client requesting access, while SV_PORT is the port number of the server. SID is a random session identifier that assists in differentiating the current request with other requests. CL_IP is the Client IP.

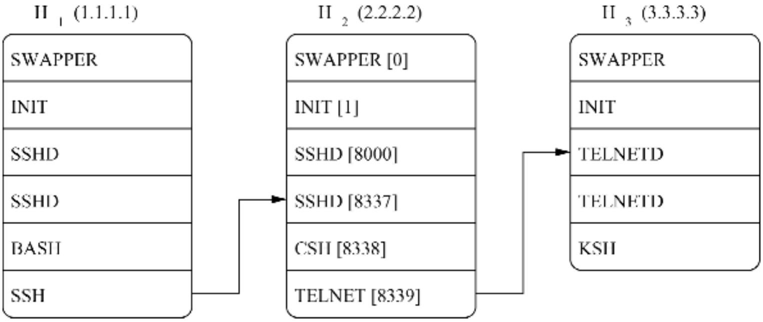


Fig. 1. Process structure of three hosts in a network loop [1]

For example, in a network loop as shown in Fig. 1, the STOP request would engage the following procedure.

1. H_3 sends a request to H_2 for the random session identifier SID along with the SV_REC.

968, 23 : SV_REC : 1029384756

2. H2 sends the SV_REC with the above SID to H_1. Through this process, information regarding the connection to Port 8337 is requested.

616, 22 : SV_REC : 1029384756

3. H1 sends the requested information to H_2. Here, the information that is to be transmitted by H_2 is appended to the information received from H_1, which is then sent to H_3.

3 Agent-Based Traceback System (ATS) Design

A Traceback Server (TS) that centrally collects all information is installed in the ATS, and a Traceback Agent Module (TAM) is run on all systems within the managed area. The TAM is initiated with the OS when each system is booted. The initial values are saved, and at the same time transmitted to the Traceback Server (TS). Whenever a new connection is generated or a modification to the system is made, the new information is sent to the Traceback Server, which is updated. In addition, the STOP response is included in the information set transmitted to TS by TAM.

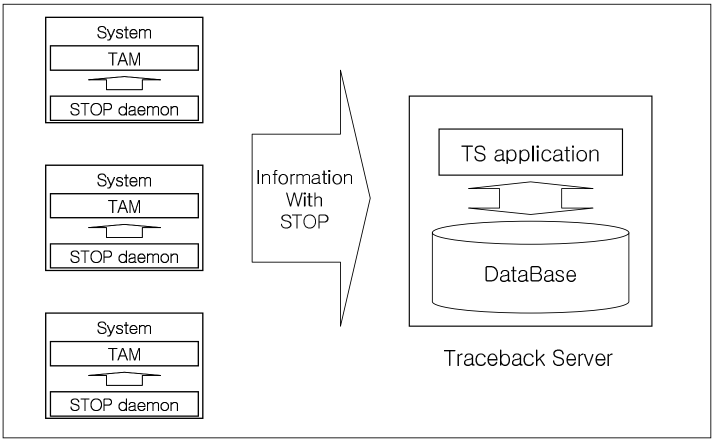


Fig. 2. overview of ATS design

3.1 Traceback Server

The TS saves the information transmitted by TAM in its internal DB. Upon the user's request, the data stored in the DB is retrieved, aiding the traceback process.

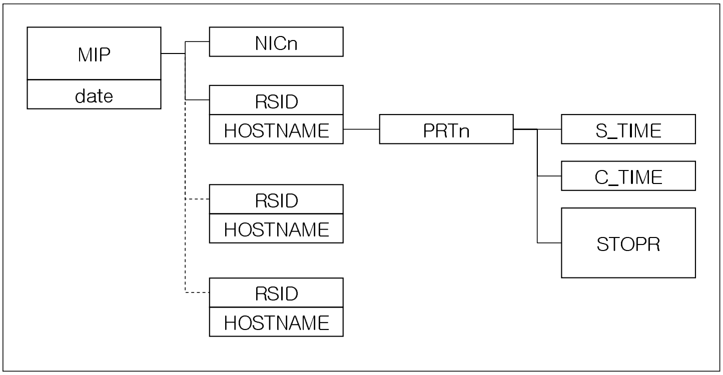


Fig. 3. Data structure saved in TS database

The length of time that the data should remain in the TS database is determined in consideration of system performance and convenience of management.

The basic structure of the TS DB is shown in Fig. 3. TS receives the MIP information sent by TAM during its initiation stage, which it generates into a new table along with the time that the information was received. Here, the NICn is also recorded. If a system is already connected at the time that TAM is initiated, the RSID and HOSTNAME, PRTn, and S_TIME are recorded as additional information. If systems with a different HOSTNAME are connected to the MIP, a new tree is generated under the MIP with the RSID and HOSTNAME of each system. Also, when a system with the same HOSTNAME terminates the connection and attempts to reconnect, a new tree is generated under the same MIP with the same HOSTNAME. This is because while the HOSTNAME remains the same, the RSID is changed. By generating a new tree, duplication of previous records is prevented.

3.2 Traceback Agent Module

The Traceback Agent Module (TAM) analyzes the network interface environment of the systems in which it is installed, and is activated by recognizing and differentiating the interfaces that are connected to the internal network from those that are not. The module collects the following information, which is sent to the TS.

Table 1. Information sent from TAM to TS

Object	Description
MIP (Managed IP address)	The managed IP addressed assigned to the system
NICn	Number of added interfaces (0,1…n)
RSID (Session ID)	A randomly generated Session ID which is used to prevent duplication when the same hostname is used to access the same port.
HOSTNAME	Hostname of the system attempting the intrusion. Returns 0 when invalid.
PRTn	The number of the Port connected to the system. Returns 0 when invalid.
S_TIME (Session start time)	The time when a new system access is attempted in each additional interface. Returns 0 when invalid.
C_TIME (Session closing time)	The closing time of the connected session. Returns 0 when invalid.
STOPR (STOP Result Forwarding)	TAM receives the information it has requested and received from the STOP, which it then transmits to the TS.

Once TAM is installed in a system, it should remain usable even if the system is moved to a different managed area, or the target TS is changed. To ensure the usability of TAM under changing circumstances, the transferred information must be standardized in the following format.

<MIP>:<NICn>:<HOSTNAME>:<PRTn>:<S_TIME>:<C_TIME>:[<STOPR>]

Once the TAM generates the information in the above format, it sends the information to the TS. Information is transferred from TAM to TS on three different occasions.

- ***When TAM is first initiated:*** Generally, the Kernel is activated when the system is first booted, which is also when the TAM is activated. The minimal information that can be transmitted at this point are the Managed IP address, and number of additional interfaces.
- ***When the Session is started:*** This is the moment when an outside access is attempted on an additionally installed interface. At this time, the Managed IP address, number of additional interfaces, HOSTNAME, PORT number, session start time and STOP information are sent.
- ***When the Session is closed:*** This is the moment when the outside access connection is closed. At this time, the Managed IP address, number of additional interfaces, HOSTNAME, PORT number, Session closing time and STOP information are sent.

In addition, this Module can be used to determine whether an additional interface has been installed on the system. By conveying this information to the Traceback Server, the Module can assist in managing the internal network.

4 Case Studies and Comparison

4.1 Case Studies

There are situations where the network topology is divided and separately managed in order to define a controlled area for administrative purposes, even if ubiquitous systems and networks are used for operational purposes. In this section, although the systems are irregularly placed, a logical line is drawn between the managed area and the outside area, and a scenario in which an outside intruder utilizes a system within the managed area to achieve access is considered.

Here, we assume that an intrusion that has been attempted in the following network has been detected by NIDS.

A is the attacker, and TAM is installed in systems B and C, which are located within the managed area. NIDS detects intrusion attempts into the internal network. A searches for a path within the managed area that can be used as a path, and discovers system B, which it uses to attack system C.

In this situation, the traceback system carries out the following tasks.

1. Because TAM has been installed in system B, the status of system B is already recorded in the DB of the TS.
2. After A gains access into system B, the connection is published, upon which the TAM sends information on the new access to the TS.

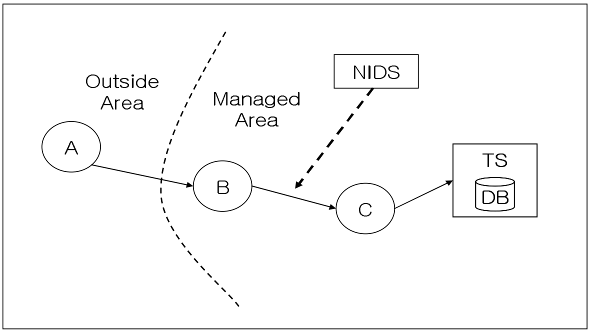


Fig. 4. Logical position of systems

3. If an attack is attempted on system C through system B, the attacked system uses STOP to request information from system B.
4. System C receives the STOP results sent from system B and sends the TAM results, which include the STOP results to the TS, where the data is saved.
5. If the intrusion is detected by NIDS or the attack is detected by system C, a signal is transmitted to the security administrator, who conducts the traceback using the information stored in the DB of the TS.

4.2 Comparison

Until now, many of the different ways of traceback method are suggested. And, each of them has their own special features.

Table 2. IP traceback methods

Object	iTrace [4]	PPM [12,13]	ATS
Network overload	High	High	Low
Bandwidth overload	Low	Low	Low
Memory Needs	High	High	Low

PPM[12,13] method includes node sampling and edge sampling. When a packet goes through the router, the router checks the packet header. After checking packet header, router marks the unique router address on that IP header and passes marked packet to next router. In this point, router marks every packet that passes through router, the router overload occurred and network traffic jam also will be growth.

iTrace method is the different way from PPM method. This method creates the iTrace message from the router and send the iTrace message to the next router. iTrace message includes payload information of the packet, and is similar to ICMP. But as a consequence, overload generating point and bandwidth overload also exists.

And, ATS is most suitable for the ubiquitous network environment. Because, in an ubiquitous environment, we can not make a prediction about intrusion routes. Like other methods have, ATS has its own traceback solution, and also has the controlling feature for networking interfaces.

5 Conclusion

In this study, ATS is proposed to trace illegitimate access that has been attempted through a path originally unintended by the internal network. By using TAM to attach additional information to the basic information provided by STOP, more detailed information is provided to aid the traceback process.

However, weaknesses can be identified in the proposed ATS. First, if the system which is activating either the TAM or the STOP daemon is compromised, this can have an impact on the individual TAM or STOP that comprise the system, which in turn can ultimately affect the entire ATS. However, because the manipulation of TAM and the STOP daemon are issues that should be dealt with under the subject of systems security, such issues were not considered in the proposed architecture. Second, if the network can be activated by an intrusion that does not pass through a general network interface but utilizes other paths such as non-network interfaces (ex; IrDA or Bluetooth), measures to respond to these attacks also need to be developed.

In actuality, it is almost impossible to draw lines between managed and unmanaged areas within a ubiquitous environment. However, the ATS architecture proposed in this study provides a means to respond flexibly to a changing topology.

In addition, the proposed ATS can be used to establish a forensics system using traceback in Enterprise Security Management (ESM).

References

1. Brian Carrier, Clay Shields: The Session Token Protocol for Forensics and Traceback. ACM Transactions on Information and System Security, Vol. 7, No. 3, (2004)
2. K. Park and H. Lee: On the effectiveness of probabilistic packet marking for IP under denial of service attack. In Proc. IEEE INFOCOM '01, pages 338-347 (2001)
3. D. X. Song, A. Perrig: Advanced and Authenticated Marking Scheme for IP Traceback. Proc. Infocom, vol. 2, pages 878-886 (2001)
4. Steve Bellovin, Tom Taylor: ICMP Traceback Messages. RFC 2026, Internet Task Force. (2003)
5. Stefan Savage, David Wetherall, Anna Karlin and Tom Anderson: Practical Network Support for IP Traceback. Technical Report UW-CSE-2000-02-01, Department of Computer Science and Engineering, University of Washington
6. R. Stone: CenterTrack: an IP overlay network for tracking DoS floods. Proc. 9th Usenix Security Symp. (2000)
7. A.C. Snoeren, C. Partridge, L.A. Sanchez, W.T. Strayer, C.E. Jones, F. Tchakountio and S.T. Kent: Hash-Based IP Traceback. BBN Technical Memorandum No. 1284 (2001)
8. H.Y. Chang et al: Deciduous : Decentralized Source Identification for Network-based Intrusions. Proc. 6th IFIP/IEEE Int'l Symp., Integrated Net., Mmgt. (1999)
9. Jung, H. T., Kim, H. L., Seo, Y. M., Choe, G., Min, S. L., Kim, C. S., and Koh, K.: Caller Identification system in the Internet environment. UNIX Security Symposium IV PRoceedings. (1993)
10. Johns, M. S.: Authentication server. RFC 931, TPSC.
11. Johns, M. S.: Identification Protocol, RFC 1413, US Department of Defense.
12. Tatsuya Baba, Shigeyuki Matsuda.: Tracing Network Attacks to Their Sources. IEEE Internet Computing. (2002)
13. Andrey Belenky, Nirwan Ansari.: On IP Traceback. IEEE Communication Magazine. (2003)

Real-Time Intrusion Detection in Ubiquitous Networks with a String-Based Approach

Bo Zhou, Qi Shi, and Madjid Merabti

School of Computing and Mathematical Sciences,
Liverpool John Moores University,
Byrom Street, Liverpool, L3 3AF, United Kingdom
B.Zhou@2004.ljmu.ac.uk, {Q.Shi, M.Merabti}@ljmu.ac.uk

Abstract. In this paper we introduce the detection details and experimental results of our proposed Service-oriented and User-centric Intrusion Detection System (SUIDS). SUIDS is designed for ubiquitous computing environments like a smart home/office. It adopts a novel auditing mechanism and flexible system architecture to meet the special requirements of ubiquitous networks. Specifically, the paper shows how a string-based method is used in a user profile to represent the user's short-term behavior in due course; and how an appropriate string length and threshold value are determined in order to balance the system's false alarm rate and detection effectiveness. As a result, SUIDS achieve real-time intrusion detection in ubiquitous networks with a lightweight and adaptable detection model.

1 Introduction

The notion of ubiquitous computing was introduced by Mark Weiser in 1991[1]. In the era of ubiquitous computing, computer-embedded devices will compose a fully connected world. These devices have the abilities to compute and communicate with each other through wired or wireless connections. Eventually it will achieve a non-intrusive availability of computers throughout physical environments.

Just like other networks, one of the main prerequisites for a ubiquitous network is adequate security [2][3]. The network has to be properly secured so that it can be relied upon. Intrusion Detection Systems (IDSs) are widely used to protect computer networks [4]. They detect and make alarms when intrusions have taken place or are taking place in the networks.

Existing IDSs have several weaknesses that hinder their direct application to ubiquitous networks. These shortcomings are caused by their lack of considerations about the heterogeneity, flexibility and resource constrains of ubiquitous networks. To overcome these issues, we proposed a Service-oriented and User-centric Intrusion Detection System (SUIDS) for ubiquitous computing environments like a smart home/office [5][6]. Briefly, in SUIDS, service-oriented event records and user profiles are used to audit users' activities and protect various networked appliances against intrusions. A user-centric approach is proposed to

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 352–359, 2006.

spontaneously compose a defence wall against malicious users. In this paper we will explain in details about the detection methods used and experiments carried out during the implementation of SUIDS. Experimental results show that SUIDS can achieve high detection efficiency while keeping the low false alarm rate. SUIDS has the following beneficial features: a reliable auditing mechanism, real-time intrusion detection technique, adaptable and heterogeneous system architecture.

The rest of the paper is organized as follows. In section 2, an overview of the system design and architecture is provided. The detection method and mathematical model used by SUIDS are described in section 3. Section 4 discusses the related experiments and results. Finally, our conclusions and future work are introduced in section 5.

2 Background and Previous Work

In paper [5] we introduced the framework of SUIDS. It is proposed to protect the heterogenous appliances in ubiquitous networks. SUIDS handles the heterogeneity issue of ubiquitous network by classifying network nodes into three major categories (head nodes, service nodes, and user nodes) and integrating intrusion detection with service specific knowledge. Head nodes such as a PC have fast connections and advanced operating systems. Service nodes such as a smart refrigerator, camera recorder and electrical door lock are used to store information or provide specific services only. User nodes are defined as those portable devices such as a user's PDA or smart phone.

Specifically, in SUIDS, the service nodes, which have received service requests from users, will dynamically form a defence wall against malicious activities. Software agents [7] on service nodes will monitor the system information across system layers, e.g. from the network layer such as an open port to the application layer such as an operation of the devices. According to the operations and system states of service nodes, event records will be generated and sent to the corresponding head nodes. Unlike existing network-based IDSs [8][9], SUIDS integrates service specific knowledge with intrusion detection and thus focuses on service level rather than the burdensome packet analysis.

As a research scenario to demonstrate the design, we assume Mike lives in a smart home. He uses his PDA to open the home door, adjust the temperature of central heating, and send documents to a printer. In SUIDS all these tasks carried by Mike or on behalf of him will be recorded and connected to his account. For example, when Mike comes into a room A, two event records will be sent to the corresponding head node:

{Mike, Door_Room_A, open, 7:28:34am}
{Mike, Door_Room_A, close, 7:28:37am, 3sec}

The '3sec' in the second event record represents the duration of this operation.

If a device has other security-related parameters, they will also be recorded. For example, when Mike uses a printer, two event records will be generated:

{Mike, Printer, print, 16pages, 11:12:23am}
{Mike, Printer, logout, 3.4Mb, 11:13:45am, 82sec}

Here '16pages' indicates that Mike has printed 16 pages of documents in this session and '3.4Mb' indicates the amount of data that has been transferred during the same session. They are all security-related parameters. Eventually, these event records will be used to create Mike's profile.

3 Detection Methods

3.1 Mathematical Model

We use the statistical component of SRI's NIDES as our mathematical model [10][11]. Instead of simply measuring the means or variances of variables, NIDES developed a more sophisticated statistical algorithm by using an X^2-like test to measure the similarity between short-term and long-term profiles. In NIDES, user profiles are represented by a number of probability density functions (PDFs). Assume S is the sample space of a random variable and events E_1, E_2,...E_k are a mutually exclusive partition of S. Let P_i denote the expected probability of the occurrence of the event E_i, P'_i denote the actual probability of the occurrence of E_i during a given time interval. The similarity between the expected and actual distributions is determined by the statistics:

$$Q = \sum_{i=1}^{k} \frac{(P'_i - P_i)^2}{P_i} \tag{1}$$

If the cumulated value of Q exceeds a pre-determined threshold during a given time interval, an alarm will be raised. To utilize this statistical component we have to define a new model to specify the service-related factors, which can effectively represent the user's both long-term and short-term behaviors.

3.2 Probability Distributions: Representation of Long-Term Behaviors

In SUIDS, a user's long-term behavior is represented by probability distributions. They indicate the possible results and corresponding probabilities of a user's each kind of action. For example, statistical results may suggest that the typical time for Mike to open his home door during a day is between 8-9am and 5-6pm. This action rarely happens during other time. Thus we can get the following probability distributions for the action of opening a home door:

{Door, open, 1-2, 3%, 8-9, 48%, 17-18, 45%, 22-23, 4%}

Where '1-2' represents the door opening time, i.e. between 1-2am; and '3%' represents the statistical probability for opening a door during this period.

Similarly, we can also represent Mike's behavior regarding his usage of a printer. Assume the recorded largest number of pages Mike had ever printed in one transaction is 200. Thus we can divide it into 10 possible groups: 1-20, 21-40,...,181-200. The occurrence probability for each group is:

$$Pi = \frac{E_i}{E} \tag{2}$$

Where E is the total number of records; E_i is the number of occurrence of the i^{th} group.

Assume the probability distributions in turn are 38%, 36%, 20%, 1%, 0%, 0%, 3%, 0%, 1%, 1%. If Mike prints 30 pages in current transaction, the similarity value is:

$$Q_i = \frac{(P_i' - 36\%)^2}{36\%} \tag{3}$$

Where P_i' denotes the actual occurrence probability of event E_i (Here it is E_2, i.e. printing 21-40 pages) during a given time interval.

Except the printed page number, other parameters such as the amount of data transferred and processing time occupied by each session are also monitored and taken into account in a similar way.

3.3 String: Representation of Short-Term Behaviors

The remaining problem now is to get the value of P_i'. Some IDSs use time interval to determine the detection window, i.e. each event only makes effect during a certain period. Because SUIDS is a distributed and mobile system, the time-based detection window will introduce the synchronization issue and make the system more complicated.

Thus in SUIDS we proposed a string-based method to determine the detection window. The 'string' is used to indicate the user's short-term behavior. For example, if the last 100 printing operations can effectively represent Mike's short-term behavior regarding his usage of the printer, a string with the length of 100 will be set to follow the printing probability distributions in his profile. Each character of the string represents one of his historical printing operations. The format of his profile becomes:

$$[\text{Printer, print, } \underbrace{1-20, 38\%; 21-40, 36\%; ...181-200, 1\%}_{\texttt{10 pairs}}. \underbrace{19082031012...15001}_{\texttt{100}}]$$

The last item here records Mike's last 100 printing operations. We use number 0-9 to represent the 10 groups, i.e. number 0 indicates printing 1-20 pages, number 1 indicates printing 21-40 pages and so on. Every time when a new record comes, the earliest record will be discarded. The value of P_i' can be calculated immediately from this string by applying equation (2).

The length of the string is variable. It depends on the system's requirement and characteristics of each event. As will be explained in the next section, longer strings may decrease the false positive rate, but at the same time the false negative rate will be increased and more system resources will be used.

4 Experiments and Results

By using the Georgia Tech Network Simulator (GTNetS) [12], we created a simulation environment to test the feasibility and applicability of SUIDS [6]. All the nodes in our simulations are connected and communicate with each other through wireless connections, i.e. in an Ad Hoc pattern. The default routing protocol is DSR [13]. Fig. 1 shows a snapshot of the simulated environment. The desktop icons represent the head nodes. The PDA icons represent the user nodes. The rest are service nodes. User nodes in our experiments are mobile. The mobility pattern is based on the Random Waypoint (RWP) model [12]. Several types of service nodes were also specified according to their traffic patterns and parameter characteristics.

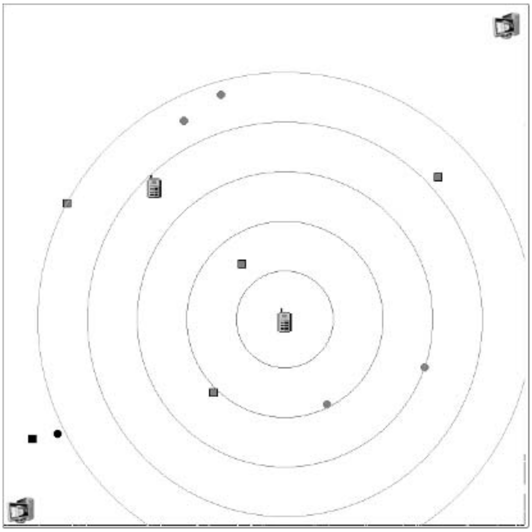


Fig. 1. A snapshot of simulated environment

The first experiment we carried out is to examine the false positive rate of SUIDS and see how the string length affects it. We set the string length from 10 to 100, respectively, and divide the audit data into two parts. The first half is used to create a user profile and the second half is used to test. Because the audit data is generated and collected under a consistent circumstance, any alarm raised during this test will be considered as a false alarm. To get a low false alarm rate, the value of Q needs to be small.

To investigate each factor's exact influence on Q, we only take the processing time into account at this stage. Other factors will be tested in our future work.

Table 1 shows the increment of Q after loading the test data into the system, with a different string length. We can see that the increment of Q decreases with the string length increasing. As expected, it indicates that a longer string is more accurate to represent the user's short-term behavior. However, because the longer string also uses more system resources, we chose the length of 80 as our investigation sample. Actually other parameters such as a threshold value also play important roles in determination of the false alarm rate.

Table 1. Increment of Q decreases with the string length increasing

Length	Q
20	89.5758
40	42.6789
60	28.7096
80	19.1924
100	13.4959

We use a set of threshold value, from 0.5 to 3.0, to calculate the system's false alarm rate. Once the cumulated value of Q exceeds the predefined threshold, an alarm will be raised and the Q will be set back to zero. The false positive rate is calculated by:

$$R_{fp} = \frac{N_a}{N_e} \tag{4}$$

Where R_{fp} is the false positive rate, N_a is the number of false alarms that have been raised, and N_e is the total number of events that have been checked. There are total 854 event records in the testing data set. The results are listed in Table 2.

Table 2. False positive rate (alarms/events). String length=80. N_e=854

Threshold	N_a	R_{fp}
0.5	32	3.75%
1.0	18	2.11%
1.5	12	1.41%
2.0	9	1.05%
2.5	7	0.82%
3.0	6	0.70%

As we can see, the false positive rate of SUIDS is quite low. Bigger threshold value shows a less 'sensitiveness' to the deviations from the user's long-term behavior. However, we cannot decide the threshold value yet as it is also related to the next experiment.

The second experiment is to examine the system's effectiveness on detecting anomalies. We generated another set of audit data. This set of data introduces anomalies or attacks by extending the processing time beyond the normal extent. The effectiveness of the system is represented by a hit rate. If an alarm is raised in connection with an event record, this record is regarded as being 'hit'. High hit rate on anomalous event records is preferred. The equation to calculate the hit rate is:

$$R_h = \frac{N_a}{N_e} \tag{5}$$

Where N_a is the number of alarms and N_e is the number of malicious events. There are total 181 anomalous records in this data set. Table 3 shows the experiments results.

Table 3. Hit rate (alarms/events). String length=80. N_e=181

Threshold	N_a	R_h
0.5	172	95.03%
1.0	165	91.16%
1.5	159	87.85%
2.0	157	86.74%
2.5	151	83.43%
3.0	149	82.32%

In most cases, the hit rate must be kept as high as possible since any ignored attack may cause serious damages to the entire system. The tolerable false alarm rate depends on the individual requirements. Normally, it is acceptable to have a false alarm rate is less than 5%. So combining Table 2 and 3, we think in this case when the threshold value is set to 0.5, SUIDS can achieve the best performance regarding both measures.

So far, the testing data we used is either pure normal or pure anomalous. More comprehensive experiments are expected with a data set combining both normal activities and anomalies. The false alarm rates and hit rates might be influenced.

5 Conclusions and Future Work

SUIDS is proposed for ubiquitous computing environments. It takes the limited capability and high heterogeneity of service nodes and high mobility of user nodes into account. In this paper, we introduced the detection details of SUIDS. It adopts a string-based method to represent a user's short-term behavior in real-time. The experimental results show that with a carefully selected string length and threshold value, SUIDS can achieve a hit rate of 95.03% with only a false alarm rate of 3.75%.

The problem with the string-based method is that it may need more system resources if the length of strings is set too long or there are too many different types of events. The size of user profiles might be too large to be transferred frequently.

Our future work includes several directions. Firstly we will focus on refining of the simulation environments. More security-related factors will be considered and tested regarding the system's detection efficiency. Secondly we will try to establish some more subtle intrusion scenarios as they will help us to test the system in depth. Last but not least, comparing with other methods such as a multivariate distance test will help to further improve the performance of SUIDS.

Acknowledgement

This research is supported in part by the UK Engineering and Physical Sciences Research Council (EPSRC) with the project reference GR/S01634/01.

References

1. M. Weiser, The computer for the 21st century. *Scientific American (International Edition), v 265, n 3*, Sept. 1991, p 66-75.
2. F. Stajano. Security for ubiquitous computing (Wiley, 2002). ISBN 0470844930.
3. H. Thompson, J. Whittaker, and M. Andrews. Intrusion detection: perspectives on the insider threat. *Computer Fraud & Security*, Jan. 2004, p 13-15.
4. H. Debar, M. Dacier, and A. Wespi. A revised taxonomy for intrusion-detection systems. *Annales des Telecommunications, v 55, n 7-8*, July-Aug. 2000, p 361-78.
5. B. Zhou, Q. Shi, and M. Merabti. A framework for intrusion detection in heterogeneous environments. *Proceedings of 3rd IEEE Consumer Communications and Networking Conference (CCNC'06), Volume 2*, Jan. 2006, Las Vegas, Nevada, USA, p. 1244 - 1248.
6. B. Zhou, Q. Shi, and M. Merabti. A novel service-oriented and user-centric intrusion detection system for ubiquitous networks. *Proceedings of IASTED International Conference on Communication, Network and Information Security (CNIS'05)*, Nov 2005, Phoenix, Arizona, USA, p. 76-81.
7. Y. Du, H. Wang, and Y. Pang. Design of a distributed intrusion detection system based on independent agents. *IEEE Proceedings of International Conference on Intelligent Sensing and Information Processing.*, 2004, p 254-7.
8. S. Northcutt and J. Novak. Network intrusion detection (New Riders Pub, 2002, c2003). ISBN: 0735712654.
9. D. Marks, P. Mell, and M. Stinson. Optimizing the scalability of network intrusion detection system using mobile agents. *Journal of Network and Systems Management, v 12, n 1*, March 2004, p 95-110.
10. T. Lunt, A. Tamaru, F. Gilham, R. Jagannathan, P. Neumann, H. Javitz, A. Valdes, and T. Garvey. A real-time intrusion detection expert system (IDES) - final technical report. *Computer Science Laboratory, SRI International*, Menlo Park, Califomia, February 1992.
11. Z. Zhang, C.Manikopoulos, J. Jorgenson. Architecture of generalized network service anomaly and fault thresholds. *MMNS 2001*: 241-255.
12. GTNetS homepage,
    http://www.ece.gatech.edu/research/labs/MANIACS/GTNetS/.
13. E. Royer and C.-K. Toh. A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks. *IEEE Pers. Commun.*, Apr. 1999, pp. 46-55.

A Security Model for Home Networks with Authority Delegation

Jin-Bum Hwang and Jong-Wook Han

University of Science & Technology, Korea
Electronics and Telecommunications Research Institute, Korea
161 Gajeong-dong, Yuseong-gu, 305-350, Korea
{hjb64253, hanjw}@etri.re.kr

Abstract. In this paper, we propose a security model that deal with the authentication and authorization problems for home networks. First, we examine existing researches for home network security and summarize their shortcomings, such as bottleneck, single point of failure, and inconvenience of configuration. Then, we introduce a new security model making up the previous works' defects. In the proposed model, we classify the services into three groups based on their security sensitivity level, and provide different security mechanism to each security level service to make a difference among the protection levels of each service (i.e. to provide more secure mechanisms to more important services.) In addition to this, we distribute the computational cost for security function to each service device while centralize the policy configuration function to central device by using authority delegation scheme. Finally, we describe how the security and convenience are enforced by using our security model. Proposed security protocols in our model are based on the SPKI/SDSI (Simple Public Key Infrastructure / Simple Distributed Security Infrastructure.) and a lightweight protocol similar to SPKI.

1 Introduction

Home network is the environment on which many intelligent devices are connected with each other and provide services which significantly increase comfort, decrease energy requirements and assists in sharing and managing resources in a residential environment [1,2.] Home networks are connected to the Internet and the services provided by home devices are closely related with the residents' privacy and safety (e.g. gas valve remote control, door lock/unlock, home viewer, home banking, and remote health care). In these circumstances, the home networks may encounter serious security problems, such as disclosing of the private information or illegal accessing to home devices and services. For example, a thief can gather information about whether there is somebody at home and can unlock the door, or someone can masquerade as a member of the home to use charged services. Therefore, the security problems should be carefully considered for home network to be widely deployed.

There were several studies on security for home networks [3,4,5,6], but most of the previous works have focused on only authentication and data encryption, and the

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 360 – 369, 2006.

access control has been largely ignored. However, the access control is one of the significant security services at home network in which various users and many kinds of services are exist. Some researches, such as UPnP middleware [7], and Echonet specification [8], include both of authentication and access control mechanism for home network services, but they also have some problems, such as bottleneck, single point of failure, and inconvenience of configuration. In this work, we present a new security model which is appropriate to home networks.

The remainder of this paper is organized as follows: Section 2 introduces the characteristics of home networks and the problems of exiting security models. Section 3 shows the architecture of proposed security model and details the procedure of proposed protocol. Section 4 discusses about how the problems are solved well and the performance analysis. Finally, we conclude our paper in section 5.

2 Background

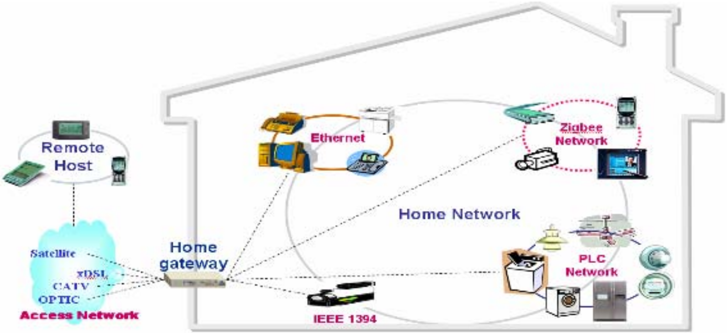


Fig. 1. The Home Network model

2.1 The Characteristics of Home Networks

Home network is connected to the Internet through variable technologies such as Hybrid Fiber/Coax (HFC) based on the existing TV cable networks, Digital Subscriber Line (DSL), Integrated Services Digital Network (ISDN) based on phone line networks, and Digital Satellite Service (DSS) based on direct broadcast satellite networks. The devices in home network are connected also through variable technologies such as Ethernet, HomePNA, IEEE802.11, Zigbee, PLC. A home gateway is an interface device that enables the variable technologies to co-exist and devices to communicate through other communication line [10,11]. Fig. 1 shows the general home network model. The devices in home network could be logically divided into three groups such as a client (C) for user to use the services, a service device (D) that provide services to client, and a home gateway (G) that mediate the communication between the client and the service device. Clients can be inside or outside of home to use the home network services.

2.2 Existing Security Models

The existing security models for home networks can be grouped into two categories, based on whether the authentication and access control decisions are made on the central device (home gateway) or on distributed devices.

Centralized security model: In the centralized security model, a client requests services to a home gateway and the home gateway authenticates the user and verifies authority of the user. Then, if it is proved that the request is from a valid user, the home gateway requests service to the service device which provides the service actually. Finally, the service device provides service to the home gateway and the home gateway relays it to the client who has requested it. [8,9]

Distributed security model: In the distributed security model, each service device authenticates the user and verifies authority of the user directly. Home gateway only connects clients and service devices physically and does not participate in the authentication and access control function. [7]

The centralized security model has several advantages that more convenient and unified security policy could be applied to home networks by using well organized technologies like Role-Based Access Control (RBAC) [12], but has shortcomings as follows.

Bottleneck: The home gateway could be a bottleneck because all requests from clients to services are authenticated and verified at the home gateway. All service requests and responses between clients and service devices are performed indirectly via home gateway, although they can communicate with each other directly.

Super User: The administrator of a home gateway can access all services that service devices provide in a home network. Therefore, the privacy of residents could be violated by the administrator.

Single point of failure: If a home gateway is compromised by an illegal user, all services at home are also compromised.

The Distributed security model has not above problems but it has also several shortcomings as follows.

Absence of unity and safety: It is hard to construct unified and safe security policies because each home resident, who is not familiar with security procedure, have to administrate the security policies of his/her own service devices.

Inconvenience: Unlike the centralized security model, all residents have to know about how to configure security policies of each services, and they have to configure the policies about dozens of service devices one by one whenever the policy change.

3 Proposed Security Model

To solve the problems of the existing security models, we introduce a hybrid security model in which a part of the accessing authority to home services are authorized by home gateway, and the others are authorized by each service devices, but the verification of all client requests are executed on each distributed service devices. We provide two kinds of protocols, one of which use asymmetric key for relatively high performance devices and the other one use symmetric key for low performance devices.

3.1 Service Classification

In our model the services, which the service devices provide, are classified into three levels as follows based on the importance of security.

Critical security: the services that closely related with the service device owner's privacy belong to this level. It is not permitted even to the home gateway administrator. Only the owner of the service device and a few users, who are allowed to use the service by owner, could access these services. Even when the home gateway is compromised by an adversary, the services need to be securely protected.

Normal security: there are several secret services that the home gateway administrator may access and the authentication and access control decisions are made at home gateway. Therefore, if the home gateway is compromised by adversary, the services at this level can not be protected.

No security: the services that do not need to be protected are member of this level. No authentication and access control functions are performed for this level.

3.2 Authorization Certificate

The proposed protocol for relatively high performance devices uses SPKI certificate which is developed for authorization in distributed environment [13]. It supports authority delegation as well as authorization. It consists of following 5-tuple: issuer, subject, delegation, authorization, validity. Issuer field specifies the issuer of the certificate who has created and signed it. The issuer is represented as a public key. Subject field defines the subject of the certificate, to whom the certificate has been issued, or to whom the certificate has been delegated. It is also represented by a public key. Delegation field specifies whether the subject of the certificate could delegate the authority specified in the certificate or not. It is represented by a Boolean value, 'True' or 'False'. Authorization field specifies the authority that defines the access right of subject. It is represented by S-expression. Validity field defines certificate expiration date.

For example, if X wants to authorize Y to use and to delegate R1 and R2 services until V date, X may issue SPKI certificate as follows.

<P(X), P(Y), True, (R1,R2), V>S(X)

P(X) represents the public key of X and <M>S(X) represents the signed message M with the private key of X. In the Above example, because the delegation field of the certificate is True, Y can delegate the authority to other subjects. If Y wants to authorize Z to use R1 service, Y may issue two SPKI certificates to Z as follows.

<P(X), P(Y), True, (R1,R2), V>S(X) and
<P(Y), P(Z), False, R1, V'>S(Y)

If Z requests R1 service and sends above two certificates to X, X performs reduction rule to make concise one certificate with the two certificate from Z.

<P(X), P(Z), False,AIntersect ((R1,R2), R1), Vintersect (V, V')>

AInersect indicates the intersection of given authorities, and VInersect indicates the intersection of given validities. Note that a server, X, is the only subject that can reduce certificates.

3.3 Authentication and Access Control with SPKI Certificate

We assume that the service devices already know the public keys of the home gateway and also ones of clients who will be authorized to use Critical security level services and the home gateway knows the public keys of all enrolled clients. The public keys could be securely distributed by using ID certificate, mobile disk, physical contact, and so on.

In proposed protocol, each service device issues two kinds of SPKI certificates. One is for home gateway to make it possible that the home gateway delegate the authority about Normal security level services to clients (NS_cert). The home gateway can delegate the authority described in this certificate to client by issuing new SPKI certificate with it (NS_cert'). The other is for clients who are authorized to access Critical security level services (CS_cert). The details of the certificates are as follows.

NS_cert: <P(D), P(G), True, AuthNS, V_G>S(D)

CS_cert: <P(D), P(C), False, AuthCS', V_C>S(D)

NS_cert':<P(D), P(G), True, AuthNS, V_G>S(D), <P(G), P(C), False, AuthNS', $V_{G''}$>S(G)

P(D), P(G), and P(C) are the public keys of the service device, the home gateway, and the client respectively. AuthNS is the list of access rights to Normal security level services, and AuthNS' is a part of the AuthNS. AuthCS' is a part of the list about access rights to Critical security level services. V_G, $V_{G'}$, and V_C are the expiration dates of each certificate.

3.3.1 Initialization Phase

When a service device is installed at a home network, the service device makes and sends a ***NS_cert*** with their operation list to the home gateway. The operation list is needed to make access control policy at the home gateway.

3.3.2 Issue Phase

Users must have appropriate certificates to use home network services. In this protocol, the issue of certificate is performed at only the first time when the client which does not have valid certificates requests a service, because having certificate about rarely used services is resource waste.

1. User requests a service to a service device.
2. If the service is one of No security level services, it is directly provided without other procedures. If it is not one of No security level services, the service device requests client to present appropriate certificate for the service with nonce N_D.
3. The client searches a required certificate and if it fails, the client requests the home gateway to issue certificate for the service by sending request message and its public key
4. The home gateway examines the policy to know whether the client is valid user for the service or not. If the request is valid, the home gateway issues a certificate ***NS_cert'*** for Normal security level services allowed to the client or request ***CS_cert*** to the service device and transfer it to client for Critical security level services. The home gateway sends N_D, and private key signature with the ***CS_cert*** request message to the service device. The ***CS_cert*** have to be transferred only via the home gateway, not directly to the client because the policy in the home gateway is in charge of protecting the service device from miss configuration.

3.3.3 Exercise Phase

The client who has an appropriate certificate can use the service by sending it and signed request message. The step 1 and 2 are same as Issue Phase.

3. The client searches required certificate. If it exists, the client makes the service request message (ServReq) and signs the messages and N_D with its private key, and then sends the signed message with the certificate.
   <ServReq, N_D>S(C), certificate
4. The service device checks the signature on the request message with the public key on the subject field of the certificate, validates the certificates in the certificate chain by verifying the signature of certificates, performs reduction rule to make concise one SPKI certificate from SPKI certificate chain (in the case of NS_cert'), and compares the authority field of certificate with the ServReq message. If the request is valid, the service device provides requested service.

3.4 Authentication and Access Control for Low Performance Devices

In this section, we propose a lightweight version of prior protocol. We assume that each service device and client share pair-wise secret key with the home gateway (K_{DG}, K_{CG}), and each service device and the clients who is allowed to use the critical services of it also share a pair-wise secret key (K_{CD}). The shared key can be distributed by various ways such as PIN number, physical contact, and so on.

We named the credential used in this protocol 'ticket' corresponding to the 'certificate.' In this protocol, there are three tickets, ***NS_ticket, CS_ticket,*** and ***NS_ticket',*** which correspond to ***NS_cert, CS_cert***, and ***NS_cert'*** respectively. The details of the tickets are as follows.

CS_ticket: <D_{ID}, C_{ID}, False, AuthCS', V_C> HMAC (K_D)

NS_ticket: <D_{ID}, G_{ID}, True, AuthNS, V_G> HMAC (K_D)

NS_ticket': <D_{ID}, G_{ID},, True, AuthNS, V_G> HMAC (K_D), <G_{ID}, C_{ID}, False, AuthNS', $V_{G'}$>HMAC (K_{DG})

D_{ID}, G_{ID} and C_{ID} are the identity of the service device, the home gateway, and the client respectively. K_{DG} is HMAC key shared between the service device and the home gateway, K_D is HMAC key used by the service device and not shared with others. K_{GC} is a shared key between the home gateway and the client, and K_{DG} is a shared key between the home gateway and the service device. AuthNS, AuthNS', AuthCS', V_G, $V_{G'}$, and V_C are all same as ones in previous SPKI certificates. <M> HMAC (K) represents signed message M by keyed hash function, and {M} K means that the message M is encrypted with key K.

3.4.1 Initialization Phase

When the service device is installed at home network, the service device makes and sends the ***NS_ticket*** with their operation list to home gateway.

3.4.2 Issue Phase

Users must have appropriate tickets to use services.

1. The client requests a service to a service device (ServReq).

   C → D : ServReq

2. If the service is one of No security level services, it is directly provided without other procedures. If it is not one of No security level services, the service device requests the client to present tickets for the service (TicketReq) and send a nonce N_D to the client.

   D → C: TicketReq, N_D

3. The client searches and generates a chain of tickets. If it fails, the client generate a nonce N_C, and requests the home gateway to issue tickets for the service by sending request message (IssueReq), its ID (C_{ID}), N_D, and N_C.

   C → G: IssueReq, C_{ID}, N_D, N_C

4. The home gateway examines the policy to know whether the client is valid user for the service or not. If the request is valid, the home gateway makes ***NS_ticket'*** for Normal security level services, generate a session key for the client and the service device (K'_{CD}). Finally it makes an encrypted massage (Message 4) and sends it to the client. The client forwards the session key information encrypted with K_{GD} to the service device.

   In the case of Critical security level services, the home gateway requests the service device to issue ***CS_ticket*** for the client by sending request message CS_ticketReq with N_D, C_{ID}, and the HMAC sign of those messages. The service device validates the request message, makes ***CS_ticket***, generates a session key (K'_{CD}), encrypts those with K_{CD} and sends them to the home gateway. And then, the home gateway transfers it to client.

   G → C: {*NS_ticket'*, K'_{CD}, $N_{C,}$ {K'_{CD}, C_{ID}, N_D} K_{GD}} K_{CG}
   C → D: {K'_{CD}, C_{ID}, N_D} K_{GD}
   ➔ In the case of Normal security level services

   G → D: <CS_ticketReq, N_D, N_C, C_{ID}> HMAC(K_{DG})
   D → G: {*CS_ticket*, $K'_{CD,}$ N_C} K_{CD}
   G → C: {*CS_ticket,* $K'_{CD,}$ N_C} K_{CD}
   ➔ In the case of Critical security level services

3.4.3 Exercise Phase

The client who has an appropriate chain of tickets can use the service by sending the chain and signed request message. The Step 1 and 2 are same as Issue Phase.

3. The client searches tickets and makes the service request message (ServReq) and signs the previous messages by keyed hash function using session key shared between it and the service device. Then, it sends the signed message with the ticket.
   C (D: <ServReq, ND> HMAC (K’CD), ticket
4. The service device checks the HMAC signature, validates the ticket in the certificate chain by verifying the HMAC signature of the ticket, and compares the authority field of the ticket with the *ServReq* message. Then, if the request is valid, the service device provides requested service

4 Discussion

4.1 Security Analysis

In this section we focus on the lightweight protocol of our security model. SPKI is a well-known technology and the security analysis of the certificate can be found in [14]. In the proposed lightweight protocol, the service device sends a nonce N_D to client when it requests the ticket for the services to the client. (Step 2. in 3.3.2) The nonce N_D is used whole of the step in issue phase and exercise phase. When the client request ticket to the home gateway, it sends N_D and a new nonce N_C with request message. Then, the home gateway encrypt the session key K'_{CD} and the nonce together, which ensure the freshness of the K'_{CD} as well as its integrity and confidentiality. It means that the protocol is secure from exposure of the old session keys. N_D is also used when the home gateway requests the service device to issue ***CS_ticket.*** (Step 4. in 3.3.2) The hash value of the request message containing N_D ensures the request is from the home gateway and not replayed. Finally, in the exercise phase (3.3.3,) the hash value of the service request message containing N_D also ensures the request is from the client and not replayed.

4.2 Concerns About Previous Works’ Shortcomings

In this section, we describe how the shortcomings of previous works are solved.

Bottleneck: In our security model, the central device (home gateway) only participates on issue phase. The client once has been issued the chain of certificates or tickets could use the service directly without home gateway until the expiration date is valid. Therefore, the bottleneck at home gateway is not occurred.

Super User: The home gateway administrator is able to access only Normal security level services and can access Critical security level services only when the service device’s owner allowed him/her.

Single point of failure: Because the access to Critical security level services is managed in each service device, Critical security level services are securely protected even if the home gateway is compromised by illegal user.

Absence of unity and safety, Inconvenience of configuration: By delegating access control management of Normal security level services to home gateway, the convenient and unified security policy could be applied to the home network. Beside this, the home gateway administrator is able to prevent the owner of the service devices from configuration mistaking about Critical security level services by not transferring the erroneous (i.e., too many authority is allowed to certain subject) certificate or ticket to client.

5 Conclusion and Future Work

In this paper, we analyzed the advantages and the shortcomings of the existing security models for home networks, and proposed a new one to solve the shortcomings but to hold the advantages of previous works. In the proposed mechanism, the services in home networks are divided into three groups, Critical security level services, Normal security level services, and No security services, based on their required security level. Only owner and a few users whom the owner authorizes can use Critical security level services which are the most important ones, and the home gateway administrator can authorizes users to access other services by constructing the security policy on home gateway and issuing the SPKI certificate or SPKI like ticket to valid clients. By using our protocols to delegate authority about Normal security level services to home gateway, the centralized security policy could be applied in home gateway but the load of authority verification could be distributed to each service device.

The home network is evolving to the ubiquitous networks. In the ubiquitous computing environment, there will be a lot of smart space domains and there also will be large amount of new security problems. The development of security framework that provides convenient user and device authentication and authorization in multi-domain environment is required for the ubiquitous services.

References

1. D. Kaleshi and M. H. Barton, “Ensuring Interoperability in a Home Networking System: A Case Study,” IEEE Trans. Consumer Electronics, Vol. 45, No. 4, Nov. 1999.
2. E.S. Eilley, “In-Home Digital Networks and Cordless Options,” IEE Colloq. On ATM in professional and consumer applications, 1997.
3. P. Krishnamurthy, J. Kabara, and T. Anusas-amornkul, “Security in Wireless Residential Networks,” IEEE Trans on Consumer Electronics, Vol. 48, No.1, Feb. 2002
4. H. Nakakita, K. Yamaguchi, M. Hashimoto, T. Saito, M. Sakurai, “A Study on Secure Wireless Networks Consisting of Home Appliances,” IEEE Trans. Consumer Electronics, Vol. 49, No. 2, May 2003.
5. A. Wacker, T. Heiber, H. Cermann, “A Key-Distribution Scheme for Wireless Home Automation Networks,” IEEE Consumer Communications and Networking Conference, Jan. 2004.

6. C. Ellison, “Interoperable Home Infrastructure – Home Network Security,” Intel Technology Journal Vol 06. Nov.2002.
7. C. Ellison, “UPnP Security Ceremonies Version 1.0,” UPnP Forum, 2003.
8. “Echonet Specification,” http://www.echonet.gr.jp
9. M. Rahman, P. Bhattacharya, “Remote access and networked appliance control using biometrics features,” IEEE Trans. Consumer Electronics, Vol. 49, No. 2, May 2003.
10. B. Rose, “Home networks, a standards perspective,” IEEE Communication Magazine, 2001.
11. S. Teger, D.J. Waks, “End-user perspectives on home networking,” IEEE Communication Magazine, 2002.
12. D.F. Ferraiolo, R. Sandhu, S. Gavrila, D.R. Kuhn, and R. Chandramouli, “Proposed NIST Standard for Role-Based Access Control,” ACM Trans. Information and System Security, Aug. 2001.
13. C. Ellison, B. Frantz, B. Lampson, R. Rivest, B. Thomas, and T. Ylonen, “SPKI Certificate Theory,” RFC2693, Sep. 1999.
14. S. Jha, T. Reps, “Analysis of SPKI/SDSI Certificates Using Model Checking,” IEEE Computer Security Foundations Workshop, June 2002.
15. M. Burrows and M. ABADI, “A Logic of Authentication,” ACM trans. Computer Systems, Vol. 8, No. 1, Feb 1990.

An Efficient Key Distribution for Ubiquitous Environment in Ad-Hoc Network Using Broadcast Encryption

Deok-Gyu Lee[1], Jang-Su Park[1], Im-Yeong Lee[1],
Yong-Seok Park[2], and Jung-Chul Ahn[2]

[1] Division of Information Technology Engineering, Soonchunhyang University, # 646,
Eupnae-ri, Shinchang-myun, Asan-si, Choongchungnam-do, Korea
{hbrhcdbr, pjswise, imylee}@sch.ac.kr
http://sec-cse.sch.ac.kr
[2] National Security Research Institute, Korea
{parkys, jcahn}@etri.re.kr

Abstract. Broadcast encryption schemes are applied to transmit digital information of multimedia, software, Pay-TV etc. in public network. Important thing is that only user who is permitted before only must be able to get digital information in broadcast encryption schemes. If broadcast message transfers, users who authority is get digital information to use private key given in the advance by oneself. Thus, user acquires message or session key to use key that broadcaster transmits, broadcaster need process that generation and distribution key in these process. Also, user secession new when join efficient key renewal need. In this paper, introduce about efficient key generation and distribution, key renewal method. Take advantage of two technique of proposal system. One is method that server creates key forecasting user without user's agreement, and another is method that server and user agree each other and create key. Advantage of two proposal system because uses a secret key broadcast message decryption do can and renewal is available effectively using one information whatever key renewal later.

1 Introduction

The broadcast encryption method has been recently applied to the transmission of digital information such as multimedia, software, pay TV, etc. As one of the key providing methods, the public key method uses a single group key to encode the session key and an infinite number of keys for decoding. As such, the server encodes the session key and enables each user to decode it using different keys. In the broadcast encryption method, only previously authorized users can gain access to digital information. When broadcast message is transmitted, authorized users can first decode the session key using the previously given private key and get digital information using this session key. In short, broadcast encryption involves generating, distributing, and renewing keys.

This paper introduces the method of generating, distributing, and renewing keys efficiently. The proposal uses 2 methods: (1) the server generates keys without the consent of users by anticipating users, and; (2) the server and users generate keys by

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 370 – 379, 2006.

mutual agreement. The advantage of the two proposed schemes is that the receiver can decode broadcast message using a secret key. Even if the key is renewed later, the user can efficiently renew using only a single set of information. In the Proposed Schemes, key renewal factor is added for fast key renewal. This allows easy key renewal and provides users with renewal values even in case of new subscription or withdrawal. This paper briefly introduces application methods in broadcast encryption, goes through the existing methods, and discusses each stage of the Proposed Schemes. Likewise, the protocols of each stage are explained. Proposed Schemes are also reviewed through comparison analysis between the existing methods and the Proposed Schemes. Finally, the conclusion is presented.

2 Overview of Broadcast Encryption and Ad-Hoc Network

2.1 Overview of Ad-Hoc Network

We envision ad-hoc networks to be formed by nodes without any prior contact, trust, or authority relation. This precludes any pre-distributed symmetric keys or a reliable (external) PKI supported by all nodes. We assume that all nodes are resource-constrained in energy, bandwidth, computational ability, memory, and possibly long term storage as well. We pay particular attention to energy utilization, since this is often the most severe constraint. Also, the energy required to transmit/receive messages in a wireless network can be equivalent to the energy used by several thousand cycles of the CPU. When sending messages, the energy consumption arises from the need to transmit a sufficiently powerful signal for good signal/noise ratio, while for receiving messages, the energy consumption comes from the signal processing necessary to decode a spread-spectrum signal. We also assume that nodes are mobile and that due to this and other environmental conditions the topology of the network can change frequently; thus, some nodes may be unreachable for some of the time. The nodes are also assumed to have low physical security; i.e., we assume they can easily stolen or otherwise compromised by an adversary.

2.2 The Current of State of Ad-Hoc Network Security

Ad-hoc network security research often focuses on secure routing protocols, which form an essential component of security in ad-hoc networks. However, all such routing schemes known to us neglect the other crucial challenge in ad-hoc security: key establishment and distribution. Protocols such as ARAN, Ariadne, SEAD, SPINS, and SRP all assume the pre-existence and pre-sharing of secret and/or public keys for all (honest) principals. This leaves ad-hoc key management and key distribution as a wide open problem. Intuitively, this should not be very surprising, as the distribution of keys in networks often mirrors trust (or authority) relations in the real world, and ad-hoc networks may not have any pre-existing trust relations. A new mechanism is needed that can accommodate the new trust scenarios in ad-hoc networks. Only recently have approaches for key distribution in ad-hoc networks been proposed Zhou and Haas introduce the idea of distributing a certificate authority (CA) throughout the network, in a threshold fashion, at the time of network formation. This CA would allow trust relations to be created in the network while also being resilient to some intrusions, malicious insiders, and breaks in connectivity. However, Zhou and Haas do not address the resource limitations of devices in ad-hoc networks. Public-key and threshold cryptography are (in general) computationally expensive and need to be tailored to the resources and constraints of low-power devices. A key management and distribution scheme that is efficient enough to be feasible for resource-constrained devices can provide the infrastructure needed by protocols for secure ad-hoc routing and can therefore enhance the set of services available for securing ad-hoc networks.

2.3 Application Methods

Broadcast encryption is based on two models. Although there are some differences between the applied models, each of them will be discussed.

This method involves generating/distributing keys using information between the user and server. This is similar to the existing multicast method, since the message provided is determined by the previous user group. The only difference lies in the transmitting method. The user participation time may be included in the key generating time, since it requires user participation in the process of key generation. Unlike the abovementioned method, the server in the second applied model generates keys.

The server generates keys by anticipating user participation at its own discretion. This method enables quick creation and renewal since the server generates all users' keys without their consent. In case the server becomes the target of attacks or other vicious purposes, however, it becomes very vulnerable.

3 Conventional Scheme – Narayanan

The Narayanan method suggests a practical paid TV scheme based on RSA, which has the ability to trace vicious users. The method of tracing vicious users can be carried out using the following principle:

> *When composing n number of $(t+1)$ vectors $x_1, x_2, \ldots, x_n$ with linear combination of arbitrary number of $s(<t)$ vectors, there is a high probability of finding the correct vectors used.*

3.1 Protocol of the Narayanan Scheme

Assume one contents provider broadcasting in m number of channels and n number of users. Protocol is divided into seven algorithms such as Setup, AddStream, AddUser, Broadcast, Receive, Subscribe, and Unsubscribe. Whether or not users receive channels can be displayed with Subsc and a $m \times n$ matrix. If user U_j is registered at S_i, the value of $Subsc[i, j]$ is 1.Otherwise, if the user is not registered, the value is 0.

Algorithm Setup

The contents provider generates the following variables:

When $N = pq, R, d_r \leq R\{1, 2, \ldots, \varphi(N)\}$, $1 \leq r \leq 4 + t$. P and q are larger prime numbers, and R is a random value. p , q , and d are composed as secret keys of the contents provider. In turn, the contents provider opens the public key (N).

Algorithm AddStream
The contents provider randomly choose $g_i \in Z_{N^*}$ to add new channel stream S_i to the system and sets up $Subsc[i, j]$ to set all j to 0; thus preventing the opening of the g_i value.

Algorithm AddUser
The contents provider chooses $(e_{1j}, e_{2j}, \dots, e_{(t+4)j})$, which satisfies $\sum_{r=1}^{t+4} e_{rj} d_r = R\Phi(N) + 1$.
At this time, U_j receives the decoding device (Set-Top Terminal) that stored the secret key in the safe memory. The secret key of U_j will be $(e_{1j}, e_{2j}, \dots, e_{(t+4)j})$.

Algorithm Subscribe
When user U_j subscribes to service S_i, the contents provider transmits g_i^{e1j} to U_j and changes the $Subsc\ [i,\ j]$ value to 1.

Algorithm Unsubscribe
When user U_j unsubscribe to S_i, the contents provider sets $Subsc\ [i,\ j] = 0$. Similar to the AddStream algorithm, the contents provider chooses a new g_i value and transmits g_i^{e1j} to all users who have the value $Subsc\ [i,\ j] = 1$.

Algorithm Broadcast
To transmit message M to channel stream S_i, the contents provider randomly chooses value x as a value smaller than $\Phi\ (N\ )$ and transmits encrypted data $C = (x, C_1, C_2, \dots, C_{t+4})$ as $C_1 = M^{d1} g_i^x, C_2 = M^{d2}, C_{t+4} = M^{dt+4}$.

Algorithm Receive
User U_j determines $\left(\prod_{r=1}^{t+4} C_r^{e_{rj}}\right) / g_i^{xe_{1j}}$ using secret key $(e_{1j}, e_{2j}, \dots, e_{(t+4)j})$ to decode encrypted data $C = (x, C_1, C_2, \dots, C_{t+4})$, which is transmitted to channel stream S_i. User U_j restores contents data M by going through this process.

$$\left(\prod_{r=1}^{t+4} C_r^{e_{rj}}\right) / g_i^{xe_{1j}} = M^{R\Phi(N)+1} = M \cdot$$

Problems of the Narayanan scheme
The Narayanan scheme requires the traffic of $(x, C_1, C_2, \dots, C_{t+4})$ per channel. Since traffic is related to the number of channels, increasing number of channels can also cause heavier traffic. In addition, despite managing to find traitor U_j, the contents provider has to distribute a new secret key to all subscribers again except U_j to disqualify U_j.

4 Proposed Scheme

Methods for efficient key renewal are proposed in a situation wherein existing users unsubscribe and new users subscribe. The proposal is the server generates and distributes keys for encrypted communication, anticipating users without their consent.

4.1 Overview of Proposed Schemes

This section presents an overview of the Proposed Schemes. Figure 1 is a classification of scenarios that can occur using the Proposed Schemes. The scenario is composed of the basic flow, renewal flow, new process flow, leaving flow, and flow of false user anticipation. The proposal can be classified into three large parts depending on the scenario: key generation and distribution, broadcast message generation, and key renewal. Similarly, two Proposed Schemes can be applied to the entire flow. Differences are only found in the initial key generation and distribution part through server anticipation and users; the rest proceeds in the same manner.

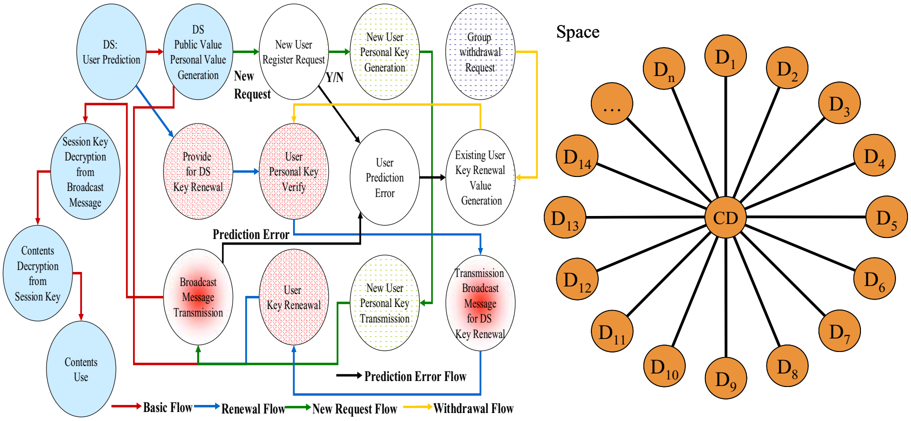

DS: User Prediction
DS Public Value Personal Value Generation
New User Register Request
New User Personal Key Generation
Group withdrawal Request
New Request
Y/N
Session Key Decryption from Broadcast Message
Provide for DS Key Renewal
User Personal Key Verify
User Prediction Error
Existing User Key Renewal Value Generation
Contents Decryption from Session Key
Prediction Error
Broadcast Message Transmission
User Key Reneawal
New User Personal Key Transmission
Transmission Broadcast Message for DS Key Renewal
Contents Use
Prediction Error Flow
Basic Flow
Renewal Flow
New Request Flow
Withdrawal Flow
Space
CD
D_1
D_2
D_3
D_4
D_5
D_6
D_7
D_8
D_9
D_{10}
D_{11}
D_{12}
D_{13}
D_{14}
…
D_n

Fig. 1. Proposed Scheme Whole Flows

In addition, the first method in the proposal has the following features: (1) the user's private key is generated by the server; (2) persons other than the user cannot decode the broadcasting message, and; (3) renewing keys is easy, which is important when new subscribers subscribe and existing users unsubscribe. On the other hand, in the second method, the user's private key is generated only upon obtaining the user's consent. When many users gather, the server generates a public key. Through the public key, the encrypted broadcasting message is transmitted. Likewise, subscribing and unsubscribing can take place easily by deleting the information provided by the user.

4.2 System Coefficient

The following is a description of the system coefficient used in this method:

p : Prime number(≥ 512 bit)

q : Prime number(≥ 160 bit $(q \mid p-1)$)

l : Number for Personal Key Generation

o : Security parameter

$d_1, \ldots, d_k$: List of Personal Decryption Key

M : Message S : Session Key

k : User e : Public Encryption Key

r_i : Set of Random Number ($r_i \in Z_p$): $(r_1, \dots, r_k)$, - $h_i = g^{r_1}$

$\langle y, h_1, \dots, h_k \rangle$: Public Key: $y = \prod_{i=1}^{k} h_i^{a_i}$ $B = M(orS)y^{aT}$, $H_i = \prod_{i=1}^{k} h_1^{a}$

$d_i = \theta_i \cdot \gamma^{(i)}$ $(\gamma^{(i)} \in \Gamma)$: $\Gamma = \gamma_1, , \gamma_\mathrm{k}$ a : Random Element($a \in Z_q$)

C : Broadcast message: $C =< M(orS)y^{aT}, h_1^a, \dots h_k^a >=< B, H_1, \dots, H_k >$

a_i : Random Number ($a_i \in Z_q$) $(a_1, \dots, a_k)$

T : Element for Key Renewal ($t_1, \dots, t_k \in Z_q$), $T = t_1 \cdot \dots \cdot t_k$

b : user's generated public information($b \in Z_p$)

ζ : User is random choose value Ξ : Stored User of ID

CD: Center Divice

4.3 Proposed Protocol

1) Key generation and distribution stage

Key generation is processed by the server. The generation and transmission of the private and public keys will go through the following process:

Step 1. The center device anticipates other devices and randomly chooses string accordingly.

$$i = 1, \dots, k \text{ prediction} \rightarrow r_i \text{ row choose} \tag{1}$$

Step 2. Based on this chosen string, the center device generates the values required to produce the public key.

$$h_i = g^{r_i} \bmod q \text{ Compute, Public Key } \langle y, h_1, \dots, h_k \rangle \tag{2}$$

$$T \text{ Generated For renewal: } T = t_1 \cdot \dots \cdot t_k$$

Step 3. The center device produces the public key using the created value h and calculates the private key.

$$\theta_i = \left(\sum_{j=1}^{k} r_j a_j t_j \right) / \left(\sum_{j=1}^{k} r_j \gamma_j \right) \bmod q \tag{3}$$

Step 4. The center device transmits the generated private key d_i to other devices.

$$d_i = \theta_i \cdot \gamma_i \tag{4}$$

Step 5. The other devices acquires θ_i from the received d_i.

$$d_i = \theta_i \cdot \gamma_i / \gamma_i \tag{5}$$

2) Center Device broadcast message generation stage

Broadcast messages can be transmitted by encrypting the session key with the encrypted message and encrypting the message itself. Both methods are described as follows:

Step 1. The center device calculates by encrypting message M or session key S .

Step 2. The center device randomly chooses factor a , operates key renewal factor T , and uses both random factor and renewal factor to produce a message.

Step 3. The center device produces and transmits the broadcast message.

$$C =< M\ (S\ )y^{\ aT}\ , h_1^{\ a}\ ,\ , h_k^{\ a}\ > \tag{6}$$

Step 4. The received message acquires message M or session key S using the private key.

$$M\ (S\ ) = C\ / U^{\ \theta_i}\ , U = \prod_{j=1}^{k} H_j^{\gamma_j} \tag{7}$$

$$U^{\ \theta_i} = \left(\prod_{j=1}^{k} H_j^{\ \gamma_j} \right) = \left(\sum_{j=1}^{k} g^{\ ar_j\gamma_j} \right)^{\theta_j} = \left(\sum_{j=1}^{k} g^{\ r_j\gamma_j} \right)^{a \cdot \theta_j} = \left(\sum_{j=1}^{k} g^{\ d_j\gamma_j} \right)^{a} = \left(\sum_{j=1}^{k} h_j^{\ d_jT} \right)^{a} = y^{\ aT}$$

$$M\ (S\ ) = M\ (S\ ) \cdot y^{\ aT}\ / y^{\ aT}$$

3) Other devices broadcast message generation stage

Broadcast messages can be transmitted by encrypting the session key with the encrypted message and encrypting the message itself. Both methods are described as follows:

Step 1. The center device calculates by encrypting message M or session key S .

Step 2. The center device randomly chooses factor A , and uses both random factor and renewal factor to produce a message.

Step 3. The other devices produces and transmits the broadcast message.

$$C =< M\ (S\ )y^{\ AT}\ , h_1^{\ A}\ ,\ , h_k^{\ A}\ > \tag{8}$$

Step 4. The received message acquires message M .

$$M\ = C\ / U^{\ \theta_i}\ , U = \prod_{j=1}^{k} H_j^{\gamma_j} \tag{9}$$

$$U^{\ \theta_i} = \left(\prod_{j=1}^{k} H_j^{\ \gamma_j} \right) = \left(\sum_{j=1}^{k} g^{\ ar_j\gamma_j} \right)^{\theta_j} = \left(\sum_{j=1}^{k} g^{\ r_j\gamma_j} \right)^{a \cdot \theta_j} = \left(\sum_{j=1}^{k} g^{\ d_j\gamma_j} \right)^{a} = \left(\sum_{j=1}^{k} h_j^{\ d_jT} \right)^{a} = y^{\ aT}$$

$$M\ = M\ \cdot y^{\ aT}\ / y^{\ aT}$$

4) Key renewal stage (center device key renewal stage)

In case of existing other devices who unsubscribe or new other devices who subscribe, the following process is carried out:

Step 1. Device i requests for withdrawal.

Step 2. The center device removes i 's renewal factor from renewal factor T to update existing other devices' private keys.

Step 3. After removal, the server renews private keys and re-transmits them to other devices.

$$\theta_i \cdot \gamma^{\ (i)} \cdot t_i^{-1} = d_i^{\ \prime} \tag{10}$$

Step 4. Users get broadcast message using the renewed keys and acquire message by decoding the encrypted message as follows:

$$M\ (S) = B\ / \ U^{\theta_i t_i^{-1}},\ U = \prod_{j=1}^{k} H_j^{\gamma_j} \tag{11}$$

Using $\left(C = \langle B, H_1, \ldots, H_K \rangle\right) = \left(C = \left\langle M\ (orS) \cdot y^{aTt_i^{-1}}, h_1^a, \ldots, h_k^a \right\rangle\right)^{\theta_i}$ compute

$$U^{\theta_i t_i^{-1}} = \left(\prod_{j=1}^{k} H_j^{\gamma_j}\right)^{\theta_i t_i^{-1}} = \left(\sum_{j=1}^{k} g^{ar_j\gamma_j}\right)^{\theta_i t_i^{-1}} = \left(\sum_{j=1}^{k} g^{r_j\gamma_j}\right)^{a\theta_i t_i^{-1}} = \left(\sum_{j=1}^{k} g^{r_j d_j t_i}\right)^{at_i^{-1}} = \left(\prod_{j=1}^{k} H_j^{d_j t_i}\right)^{a} = y^{aTt_i^{-1}}$$

$$M(S) = M(S) \cdot y^{aTt_i^{-1}} / y^{aTt_i^{-1}}$$

5 Comparison Analysis Between the Conventional Scheme and Proposed Scheme

This paper proposes the broadcast encryption method, which is more efficient than the existing method in generating and renewing keys. The stability of the Proposed Scheme is based on discrete algebra issue. Compared to the existing method, the Proposed Scheme achieves efficiency in user participation, key renewal, user withdrawal, or operating amount. In this section, the efficiency of the Proposed Scheme is presented vis-à-vis the existing method.

User participation
In the existing method, the server anticipates users, generates keys in advance without user participation, and provides and distributes them to new users who subscribe. In this method, when an attack is made on the server itself, all keys created by the server can be affected.

Key renewal
In the existing Key Pre-distribution Scheme (KPS), message is transmitted as encrypted using this scheme after the key is generated and distributed. When the session is closed after the user checks the transmitted message, a key is newly produced and transmitted. If an attack is made on the key, all keys will be re-generated instead of merely renewing them. In the Proposed Scheme, however, keys are ready to use after renewing the existing users' keys in case of subscription or withdrawal.

Re-operation due to false prediction error
In the existing method and the Proposed Scheme - I, the server should set up and control the system. If the server controls flexible users, the anticipation of users should be carried out correctly. Therefore, the server should implement re-operation or additional operation in case initial anticipation fails. In the existing method, however, there is no such operation in case of failure of user anticipation. In the Proposed Scheme, user anticipation can be achieved smoothly through a simple operation like g^r when the server configures the system. Likewise, random number r can be generated on Z_p. Problems can also be solved by giving numbers larger than the expected number of users in advance.

6 Conclusion

Broadcast encryption is used to provide contents only for authorized users on the open network. Except authorized users, nobody can obtain messages from the broadcast message; authorized users can obtain the session key, with the private key transmitted in advance. This paper proposes the method of generation, distribution, and renewal of private key and suggests an easier way of renewing after users' requests for withdrawal or process of the server's withdrawal for existing users. Further studies on user tracing and key cycling are recommended.

References

1. Amos Fiat, and Moni Naor, "Broadcast Encryption", Crypto'93, LNCS 773, 480-491
2. C. Blundo, Luiz A. Frota Mattos, D.R. Stinson, "Generalized Beimel-Chor schemes for Broadcast Enryption and Interactive Key Distribution", Crypto'96, LNCS 1109
3. Carlo Blundo, Luiz A. Frota Mattos, and Douglas R. Stinson, " Trade-offs Between Communication and Storage in Unconditionally Secure Schemes for Broadcast Encryption and Interactive Key Distribution", Crypto 98
4. Juan A. Garay, Jessica Staddon, and Avishai Wool, "Long-Lived Broadcast Encryption", Crypto'00, LNCS 1880, 333-352
5. Ignacio Gracia, Sebastia Martin, and Carles Padro, "Improving the Trade-off Between Storage and Communication in Broadcast Encryption Schemes", 2001
6. Dani Halevy, and Adi Shamir, "The LSD Broadcast Encryption Scheme," Crypto'02, LNCS 2442, 47-60
7. Yevgeniy Dodis and Nelly Fazio, "Public Key Broadcast Encryption for Stateless Receivers", DRM2002, 2002. 11. 18
8. Donald Beaver, and Nicol So, "Global, Unpredictable Bit Generation Without Broadcast," 1993
9. Michel Abdalla, Yucal Shavitt, And Avishai Wool, "Towards Marking Broadcast Encryption Practical", FC'99, LNCS 1648
10. Dong Hun Lee, Hyun Jung Kim, and Jong In Lim, "Efficient Public-Key Traitor Tracing in Provably Secure Broadcast Encryption with Unlimited Revocation
11. A. Narayanan, "Practical Pay TV schemes," to appear in the Proceedings of ACISP03, July, 2003
12. R. B. Bobba, L. Eschenauer, V. Gligor, and W. A. Arbaugh.Bootstrapping Security Associations for Routing in Mobile Ad-Hoc Networks. Technical Report, Institute for Systems Research, UMd, TR 2002-44, 2002.
13. A. Boldyreva. Threshold Signatures, Multisignatures and Blind Signatures Based on the Gap-Diffie-Hellman-Group Signature Scheme. In International Workshop on Practice and Theory in Public Key Cryptography, January 2003.
14. D. Boneh and M. Franklin. Identity-Based Encryption from the Weil Pairing. In J. Killian, editor, Advances in Cryptology, CRYPTO 2001, volume 2139 of Lecture Notes in Computer Science, pages 213–229. Springer Verlag, August 2001.
15. D. Boneh, B. Lynn, and H. Shacham. Short signatures from the Weil pairing. In C. Boyd, editor, Advances in Cryptology, ASIACRYPT 2001, volume 2248 of Lecture Notes in Computer Science, pages 512–532. Springer Verlag, 2001.

16. J. C. Cha and J. H. Cheon. An Identity-Based Signature from Gap Diffie-Hellman Groups. In International Workshop on Practice and Theory in Public Key Cryptography, January 2003.
17. B. Dahill, B. Levine, E. Royer, and C. Shields. A Secure Routing Protocol for Ad Hoc Networks. Technical Report UM-CS-2001-037, University of Massachusetts, August 2001.
18. R. Gennaro, S. Jarecki, H. Krawczyk, and T. Rabin. Secure Distributed Key Generation for Discrete-Log Based Cryptosystems. Eurocrypt, 1999.
19. Y.-C. Hu, D. Johnson, and A. Perrig. SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks. In Workshop on Mobile Computing Systems and Applications. IEEE, June 2002.
20. Y.-C. Hu, D. B. Johnson, and A. Perrig. Secure On-Demand Routing Protocols in Ad Hoc Networks. Unpublished, 2001.
21. A. Khalili and W. A. Arbaugh. Security of wireless ad-hoc networks. Work in progress, http://www.cs.umd.edu/˜aram/wireless/survey.pdf, 2002.

Distributed Certificate Authority Under the GRID-Location Aided Routing Protocol

JiHyung Lim[1], DaeHun Nyang[1,*], Jeonil Kang[1],
KyungHee Lee[2], and Hyotaek Lim[3]

[1] Information Security Research Laboratory,
INHA University, Korea
{bolter, dreamx}@seclab.inha.ac.kr, nyang@inha.ac.kr
[2] Department of Electrical Engineering, University of Suwon, Korea
khlee@suwon.ac.kr
[3] DongSeo University, Korea
htlim@kowon.dongseo.ac.kr

Abstract. Ad hoc network is the network which can be considered without a pre-constructed infrastructure, and a mobile node can join the network freely. However, the participation of the mobile nodes to the ad hoc network brings up much burden of re-computation for new routes, because it leads to losing the connection frequently. Therefore, it needs authentication against the mobile nodes. To make that possible, we have two methods: single Certificate Authority(CA) and distributed CA. In the case of single CA method, the wireless network can be collapsed owing to expose the CA, but still the distributed CA method is a little safer than previous one because it needs attacks toward a lot of CAs to collapse the network. We can consider secret sharing scheme as the method that constructs the distributed CA system, but it is weak when the network size is too large. In this paper, we suggest hierarchical structure for the authentication method, and show the results of simulation.

Keywords: Ad Hoc Network Security, Authentication, Certificate Issuing.

1 Introduction

Various researches have been considered to solve the ad hoc network security problems, and most of the studies focus on how to exchange key between nodes, and how to transmit the data though secure paths[6][7]. Also, threshold cryptography has been adopted to construct the "distributed CA" by distributing private keys to several nodes[2]. In this case, public key information must be signed by at least threshold numbers of partial keys and then, they must be combined to make a valid certificate for some global public key. However, if this model is adapted, it might lead to some problems in large-scale network, because it brings up an amount of overhead to request partial certificates from whole nodes in the network.

[*] Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 380–387, 2006.

In this paper, we suggest an efficient method to implement distributed CA using the hierarchical grid structure. When a node participates in the network, it is issued a certificate by nodes in the smallest grid, not in the whole network. To transmit data to the other nodes, a node checks whether the target node is in its group(grid) or not. If not, the node should increase grid level to the smallest level where these two nodes reside in that same grid. It would be more efficient than certificate issuing method by the whole network because we can take advantage of locality in communication. We will describe this method in section 2 and prove the efficiency by simulation in section 3.

2 Proposed Authentication Protocol

The basic routing technique which we used in our scheme is Grid Location Service[1]. We add some processes to perform the certificate issuing function. A node needs its own public and private key pair, and has to obtain the information of the group which it is going to join in from its neighbor nodes before it participates in the network. If the nodes know the information about dealer node, they can obtain a group list and share values from a dealer node. The node should obtain signed public key information as a certificate from the nodes in its group list before it sends out some data. If the destination node does not belong to its group, the node should obtain larger group's node list for communication such that both the source and the destination should belong to the same group.

2.1 Packet Type and Share Holding Table

Some new packet types are needed for performing certificate issuing process in the existing grid location service scheme [1]. Packet types can be classified by type for obtaining group share or certificate.

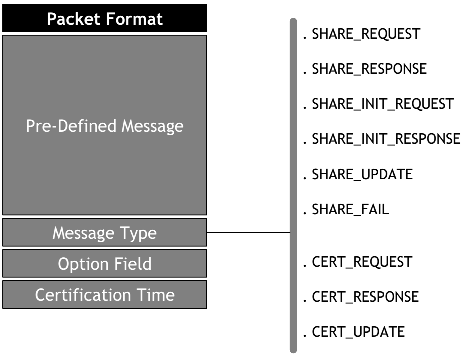


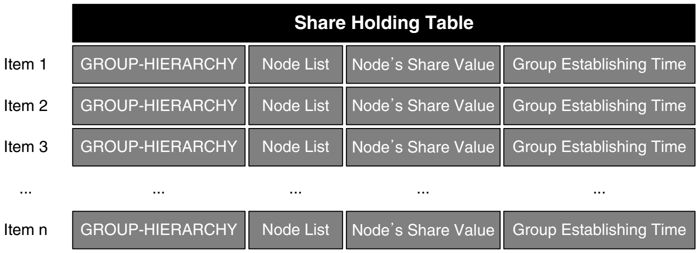
Fig. 1. Packet Format

The Packet for Obtaining Group Share

- SHARE_REQUEST: the message for requesting the information including share about its group from neighbor nodes.
- SHARE_RESPONSE: the message for responding to SHARE_REQUEST message containing Dealer ID.
- SHARE_INIT_REQUEST: the message for requesting establising a group with neighbor nodes.
- SHARE_INIT_RESPONSE: the message for responding to SHARE_INIT_REQUEST.
- SHARE_UPDATE: the message for sending the changes of a group to the group members.
- SHARE_FAIL: the message for notifying that a node has no information to SHARE_REQUEST message.

The Packet for Obtaining Certificate

- CERT_REQUEST: the message for requesting a signed message to a node's certificate from its group members.
- CERT_RESPONSE: the message for responding to CERT_REQUEST and including partially signed message.
- CERT_UPDATE: the message for notifying the update of expired certificate.



	Share Holding Table			
Item 1	GROUP-HIERARCHY	Node List	Node´s Share Value	Group Establishing Time
Item 2	GROUP-HIERARCHY	Node List	Node´s Share Value	Group Establishing Time
Item 3	GROUP-HIERARCHY	Node List	Node´s Share Value	Group Establishing Time
...	...	...	...	...
Item n	GROUP-HIERARCHY	Node List	Node´s Share Value	Group Establishing Time

Fig. 2. Share Holding Table

Share Holding Table. Every node needs the structure called "Share Holding Table" like in figure 2 for keeping secret shares of each group. "GROUP-HIERARCHY" is a group's ID that is based on x and y axis and is unique value. "Node List" contains nodes' IDs that belong to the order indicated by the value of GROUP-HIERARCHY. Each node can request a certificate from the group members, and then it can communicate to the others using that value. "Node Share Value" can be taken from the dealer node of a group, and a node uses it when the others request signed messages. "Group Establishing Time" is the time value when the owner of the table has joined in the corresponding group and received the share.

2.2 Protocol

At the beginning, a node should prepare its own public and private key pair to use it in the network. Our protocol consists of two phases. The first phase takes place when a node obtains the share value of the group, and the second phase takes place when the node obtains the certificate from the other nodes using the value. A node should obtain the certificate of the group for communicating with the other nodes, and the certificate also must be recognizable by the destination node. To do this, a node performs the following share acquisition procedure and the certificate acquisition procedure for the grid of order 1 whenever it joins the network. Also, when it sends out some data and the destination node does not belong to its group, the source node should obtain larger group's node list for communication such that both the source and the destination should belong to the same group. In the worst case, the common group might be the world grid. In the following protocol description, the dealer node takes part in the protocol, but the dealer node can be removed from the protocol because the dealer can be distributed over all the nodes if the secret sharing scheme is homomorphic.

Share Acquisition Procedure. A node should obtain the information of the group that it is going to join in before being issued a certificate. The information can be inserted into Share Holding Table of the node. The share acquisition procedure is as following :

A sender decides the order's #Order so that both sender and receiver may belong to the same grid of #Order. For obtaining the information about the group, the node broadcasts SHARE_REQUEST message to neighbor nodes. Referring the Share Holding Table, they respond with SHARE_RESPONSE if they have the information about their group, or respond with SHARE_FAIL if they do not have. If the node receives SHARE_RESPONSE message, it should register itself to its group's dealer node. And then, the node can obtain group list and share value from SHARE_UPDATE message from the dealer. If the node receives SHARE_FAIL message, it has to send SHARE_INIT_REQUEST message again for establishing a group with neighbor nodes. If a node receives SHARE_INIT_REQUEST message, it should relay that message to neighbor nodes and make the neighbor nodes send SHARE_INIT_RESPONSE message to the first issuer of SHARE_INIT_REQUEST. By sending SHARE_UPDATE to all nodes of concerned group, a dealer node should update their Node List when a new node arrives.

Certificate Acquisition Procedure. After the share acquisition procedure, a node can do certificate acquisition procedure as in the following : If a node receives the information about a group, it should obtain the certificate which is used in the group. The node sends the message including its own public key to the other nodes in the group as CERT_REQUEST message. If the other nodes receive that message, they should sign the message by their partial private key for the group and return it as CERT_RESPONSE message. If a certificate is expired after the node obtains the certificate, the node should notify neighbor nodes that the node needs new certificate by CERT_UPDATE. By sending

CERT_UPDATE, a node should update its certificate from the node of the concerned group when the time of the certificate is expired. By performing the share acquisition procedure and the certificate acquisition procedure, the sender can obtain a certificate that is verifiable by the receiver.

3 Simulation and Results

We used the discrete network simulation tool called NS-2 for our simulation and the scenarios of the simulation have two cases: non-mobility and mobility. We compared with two methods, authentication method by whole network(named "Entire") and by group network(named "Grid") in each scenario. "Grid 1" is the case when the nodes communicate in order 1. "Grid 2" is the case when the nodes communicate in order 1(50%) and order 2(50%). "Grid 3" is the case when the nodes communicate in order 1(30%), order 2(30%), and order 3(40%). "Random" is the case when the nodes randomly communicate in all orders.

We performed this simulation using 100 mobile nodes for 400 seconds.

3.1 Non-mobility Case

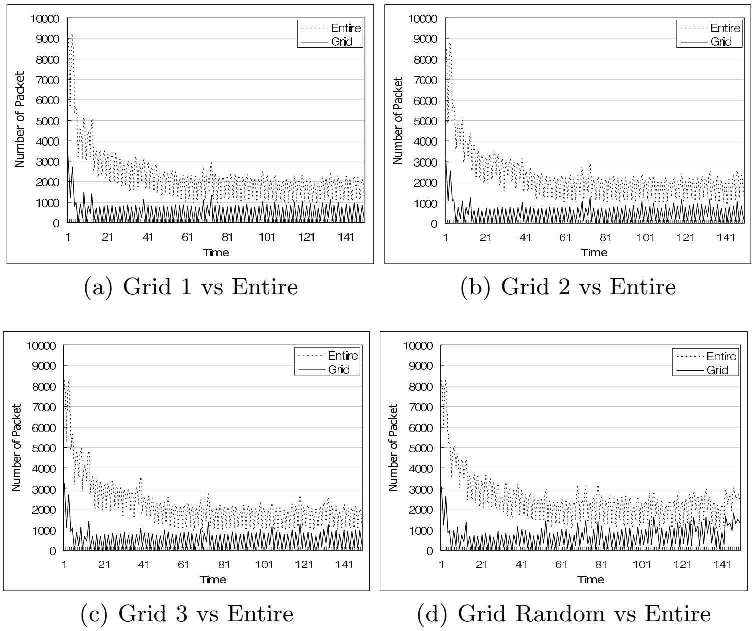
Total Control Packet. The amount of total packets is shown in figure 3. A lot of control packets for issuing a certificate are required to issue a certificate in the beginning of Entire case because share acquisition and certificate acquisition procedures are required before the first transmission of data. On the other hand, small numbers of packets in Grid method are required because only the nodes in the grid that covers both sender and receiver involve in the certificate issuing procedure. Also, Grid method consumes shorter time than Entire method before transmitting data after certificate issuing.

Control Packets for Certificate. The amount of control packets for issuing certificate per unit time (second) is shown in figure 4. After a certificate issuing is completed in the beginning, the nodes keep generating the packets for updating certificates. Grid method requires short time, about 5 $\sim$ 6 seconds, but the Entire method requires longer time about 30 $\sim$ 40 seconds for initial certificate issuing.

Accumulated Total Packets. Figure 5 shows the number of accumulated total control packets in the time. We could notice the big difference in the amount of packets used in Grid and Entire method from the figure.

3.2 Mobility Case

Threr is no big difference between the Non-mobility and mobility cases, even though the result was different from our expectation. This unexpectation might be because of following reasons : nodes generate a lot of packets to be issued a certificate at the beginning because they have no information about the groups, but a lot of nodes which obtain the information can serve it to the other nodes. Therefore, we might be able to say that the mobility of nodes hardly have an influence on the amount of the total control packets of all nodes.



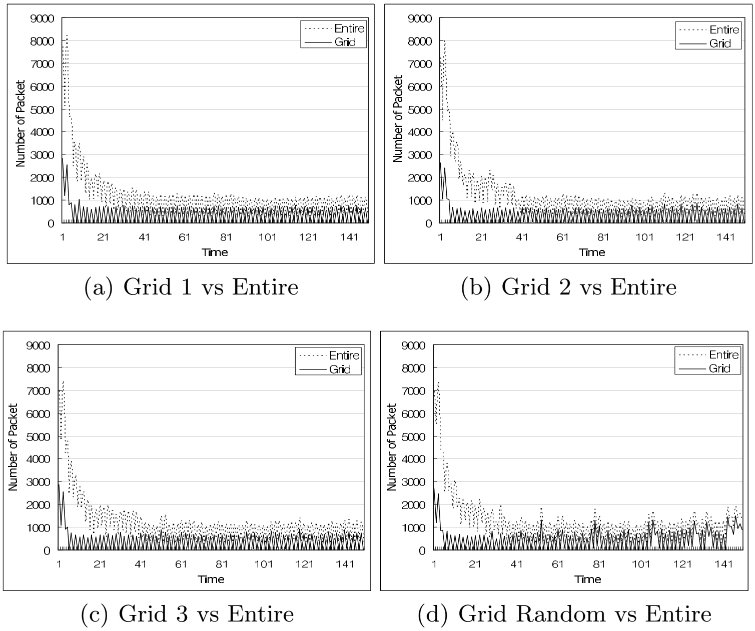
(a) Grid 1 vs Entire

(b) Grid 2 vs Entire

(c) Grid 3 vs Entire

(d) Grid Random vs Entire

Fig. 3. Total Control Packets



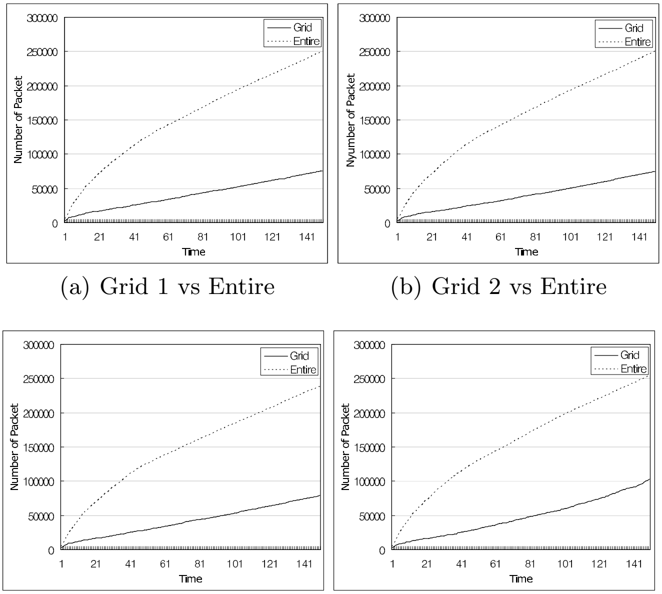
(a) Grid 1 vs Entire

(b) Grid 2 vs Entire

(c) Grid 3 vs Entire

(d) Grid Random vs Entire

Fig. 4. Control Packets for issuing a certificate



(a) Grid 1 vs Entire

(b) Grid 2 vs Entire

(c) Grid 3 vs Entire

(d) Grid Random vs Entire

Fig. 5. Accumulated Total Packet

3.3 Certificate Issuing Latency

Certificate Issuing latency is the time to be measured for taking certificate before data transmission. It is from the moment when a node requests a group share till a node obtains messages signed by partial private keys from over 50% of nodes that have received requesting messages. This result shown in figure 6 seems to be based on two reasons. At first, it has difference because the time for a node to obtain a group share is faster than the time required to obtain a certificate. Secondly, if dealer node is changed while a node is obtaining a certificate, the node should obtain a new group share from changed dealer node and it should perform the certificate issuing procedure again.

4 Conclusion

In this paper, we discussed how to construct a distributed CA efficiently using grid location service scheme. Existing the secret sharing methods can construct a distributed CA using a partial private key in ad hoc network, but it has a problem in that it costs long delay for issuing a certificate. It brings up the result that it wastes network resources, because a node should involve in certificate issuing procedure even though it does not want to communicate with others.

We solved that problem using a hierarchical grid for constructing a distributed CA. Using our scheme, delay and traffic can be shorter and less than the entire

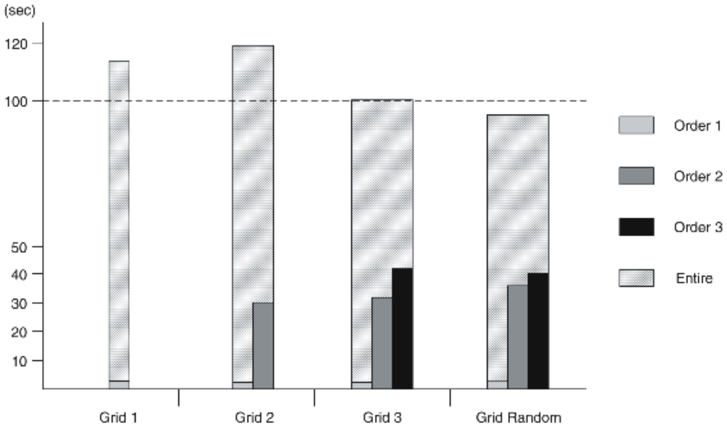


Fig. 6. Average certificate issuing latency of Entire and Grid method(unit: *sec*)

method in the beginning of the network operation. We expect that our scheme can be applied as a certificate issuing framework for other different wireless network systems.

References

1. Jinyang Li: A Scalable Location Service for Geographic Ad Hoc Routing, 6th ACM International Conference on Mobile computing and Networking (MobiCom'00), pp.120-130, Aug. 2000
2. L. Zhou and Z.J. Haas: Securing Ad Hoc Networks, IEEE Network Magazine, Vol. 13, no. 6, Nov. 1999
3. Jiejun Kong: Providing Robust and Ubiquitous Security Support for Mobile Ad-Hoc Networks, IEEE 9th International Conference on Network Protocols (ICNP'01), pp.251-260, 2001
4. Shamir A. : How to Share a Secret, Communication of the ACM, pp.612-613, Nov. 1979
5. T.P. Pedersen : A threshold cryptosystem without a trusted party, Advances in Cryptology, Proceedings of the CRYPTO'91, pp. 522-526, 1991
6. Stephen Carter and Alec Yasinsac, Secure Position Aided Ad hoc Routing Protocol, Proceedings of the IASTED International Conference on Communications and Computer Networks (CCN02) Nov 3-4, 2002
7. Jean-Pierre Hubaux, Levente Buttyan, Srdjan Capkun, The Quest for Security in Mobile Ad Hoc Networks, Proceedings of the 2001 ACM International Symposium on Mobile ad hoc networking & computing 2001, 2001

An Efficient Hierarchical Group Key Management Protocol for a Ubiquitous Computing Environment*

Sangjin Kim[1], Taewook Ahn[2], and Heekuck Oh[2]

[1] Korea University of Technology and Education,
School of Internet Media Engineering, Republic of Korea
sangjin@kut.ac.kr
[2] Hanyang University, Department of Computer Science and Engineering,
Republic of Korea
{twahn, hkoh}@cse.hanyang.ac.kr

Abstract. We propose a new centralized hierarchical group key management protocol for a ubiquitous computing environment. The proposed scheme only uses XOR and hash operations during group key updates. Moreover, the order of the total message size sent during key updates is $O(\log n)$, where n is the size of the group. As a result, our scheme is very practical, scalable, and well suited for low-power mobile devices. We have also proved the security of our proposed protocol.

Keywords: group key management, centralized hierarchical scheme, ubiquitous computing.

1 Introduction

If a user wants to transmit an identical message to n users, the user can significantly save the network bandwidth by using the IP multicast. In this case, a single copy of the message is transmitted instead of n copies of the same message. However, if a user has to securely transmit an identical message to n users, a single cryptographic key must be shared between n users to take advantage of IP multicast. Various secure group communication mechanisms have been introduced to solve this problem [1]. These mechanisms can be categorized into centralized [2, 3], decentralized [4], and distributed schemes [5], depending on whether a single trusted server, multiple servers, or none is/are used to manage the group key, respectively. Most of the works have concentrated on solving dynamic membership change that preserve forward and backward secrecy efficiently. Most of them, however, do not consider the limitation of mobile computing environment. Forward secrecy means that members withdrawn from the group cannot obtain future group keys. Backward secrecy means that newly joined members cannot obtain past group keys.

Recent works on group key management are more focused on distributed schemes. However, in a ubiquitous environment, centralized schemes are more appropriate than

* This work was supported by the Ministry of Information and Communication, Korea, under the HNRC-ITRC program supervised by the IITA.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 388–395, 2006.

distributed schemes because of the cost involved. For example, TGDH scheme [5], which is a typical distributed scheme, requires a vast number of exponentiations. Recently, Bresson et al. have introduced a centralized scheme suitable for low-power mobile devices [6]. However, it requires n unicasts each time a user joins or leaves. In this paper, we propose a new centralized hierarchical group key agreement protocol that is more efficient and scalable than Bressen et al.'s scheme with respect to network bandwidth usage. We have reduced the order of the size of the multicast messages sent during key updates from $O(n)$ to $O(\log n)$. Moreover, our scheme does not require any public/secret key operations during key updates. Our scheme only uses XOR and hash operations.

The rest of the paper is organized as follows. In section 2, we will review some related works. In section 3, we will introduce our scheme in detail and full analysis of our scheme will be given in section 4. Finally, we will conclude our work in section 5.

2 Related Work

In this section, we will briefly review centralized hierarchical schemes [2,3] and Bresson et al.'s scheme [6]. In centralized hierarchical schemes, a single trusted server is used to manage the group key. We will call this server GKM (Group Key Manager). Generally, centralized hierarchical schemes use logical binary key tree. There is a key associated with each node in the tree. The key associated with the root node is used as the group key. Each member is associated with a leaf node of the tree. In LKH scheme [2], each member has to maintain keys associated with each node along the key path. Key path is the set of nodes along the path from a leaf node to the root. In OFT scheme [3], each key associated with internal nodes is computed using the left(K_L) and the right(K_R) child key as follows: $K = H(H(K_L)||H(K_R))$, where $H(K_X)$ is the blind key of the key K_X and H is a collision-resistant hash function. In this scheme, each member maintains its key and the blind keys of its co-path. Co-path of a leaf node is the set of sibling nodes of the nodes included in its key path. In both schemes, the keys

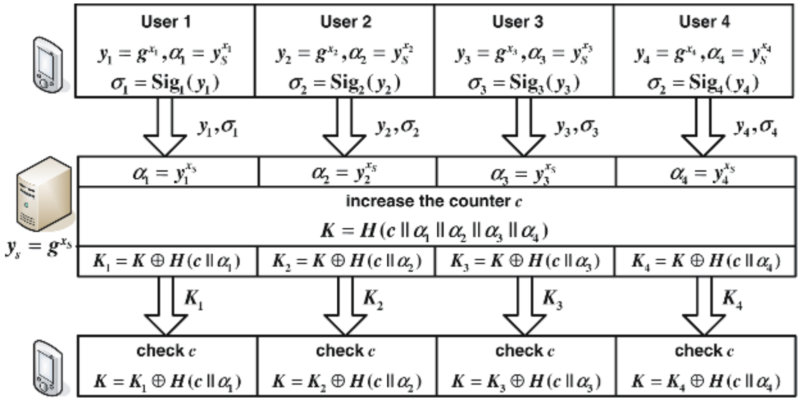

User 1
$y_1 = g^{x_1}, \alpha_1 = y_S^{x_1}$
$\sigma_1 = \mathrm{Sig}_1(y_1)$
User 2
$y_2 = g^{x_2}, \alpha_2 = y_S^{x_2}$
$\sigma_2 = \mathrm{Sig}_2(y_2)$
User 3
$y_3 = g^{x_3}, \alpha_3 = y_S^{x_3}$
$\sigma_3 = \mathrm{Sig}_3(y_3)$
User 4
$y_4 = g^{x_4}, \alpha_4 = y_S^{x_4}$
$\sigma_2 = \mathrm{Sig}_4(y_4)$
y_1, σ_1
y_2, σ_2
y_3, σ_3
y_4, σ_4
$\alpha_1 = y_1^{x_S}$
$\alpha_2 = y_2^{x_S}$
$\alpha_3 = y_3^{x_S}$
$\alpha_4 = y_4^{x_S}$
increase the counter c
$K = H(c \| \alpha_1 \| \alpha_2 \| \alpha_3 \| \alpha_4)$
$y_s = g^{x_S}$
$K_1 = K \oplus H(c \| \alpha_1)$
$K_2 = K \oplus H(c \| \alpha_2)$
$K_3 = K \oplus H(c \| \alpha_3)$
$K_4 = K \oplus H(c \| \alpha_4)$
K_1
K_2
K_3
K_4
check c
$K = K_1 \oplus H(c \| \alpha_1)$
check c
$K = K_2 \oplus H(c \| \alpha_2)$
check c
$K = K_3 \oplus H(c \| \alpha_3)$
check c
$K = K_4 \oplus H(c \| \alpha_4)$

Fig. 1. Bresson et al. scheme

or the blind keys changed during key updates were sent encrypted. These centralized hierarchical schemes require a multicast message of size in $O(\log n)$ to update group keys, where n is the size of the group.

In Bresson et al.'s scheme, which is depicted in Fig 1, each user i chooses a private key $x_i \in \mathbb{Z}_q^*$, and computes the corresponding public key $y_i = g^{x_i}$ before joining the group, where g is a generator of multiplicative group of prime order q. These keys are only used in computing the group key. Each user i sends y_i and $\mathrm{Sig}_i(y_i)$ to the GKM, where $\mathrm{Sig}_i(M)$ denotes user i's signature on M. The GKM verifies signatures of users and computes $\alpha_i = y_i^{x_S}$, where x_S is the private key of the GKM. The GKM then computes the group key $K = H(c||\alpha_1||\ldots||\alpha_n)$, where c is a counter and n is the total number of users. The counter c is incremented each time the group key is updated. The GKM sends $K_i = K \oplus H_1(c||\alpha_i)$ to each user. Each user can obtain the group key since they can compute $\alpha_i = y_S^{x_i}$, where $y_S = g^{x_S}$ is the public key of GKM. This process is repeated each time a user joins or leaves the group. As a result, Bresson et al.'s scheme always requires n unicasts to update the group key.

3 Our Group Key Agreement Scheme

Our scheme is centralized hierarchical scheme which combines ideas from centralized key hierarchical schemes and Bresson et al.'s scheme.

3.1 System Setup

We assume that GKM shares a symmetric key with each members. We will denote K_i as the key shared between GKM and user i. We assume that GKM securely authenticates each user before allowing the user to join the group. During this process, the user

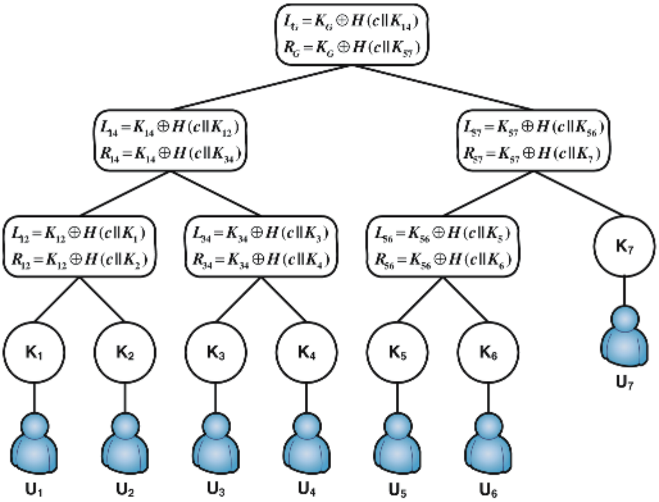

$L_G = K_G \oplus H(c||K_{14})$
$R_G = K_G \oplus H(c||K_{57})$
$L_{14} = K_{14} \oplus H(c||K_{12})$
$R_{14} = K_{14} \oplus H(c||K_{34})$
$L_{57} = K_{57} \oplus H(c||K_{56})$
$R_{57} = K_{57} \oplus H(c||K_7)$
$L_{12} = K_{12} \oplus H(c||K_1)$
$R_{12} = K_{12} \oplus H(c||K_2)$
$L_{34} = K_{34} \oplus H(c||K_3)$
$R_{34} = K_{34} \oplus H(c||K_4)$
$L_{56} = K_{56} \oplus H(c||K_5)$
$R_{56} = K_{56} \oplus H(c||K_6)$
K_7
U_7
K_1
K_2
K_3
K_4
K_5
K_6
U_1
U_2
U_3
U_4
U_5
U_6

Fig. 2. Initial Logical Key Tree

establishes a long-term key with the GKM. Our scheme does not use any asymmetric or symmetric key operation during group key updates. However, our system may require means, that may include public key operations, to authenticate users before allowing them to join the group. The mechanism used for this purpose is omitted in this paper. Initial group key is computed as follows.

- **Step 1.** The GKM constructs a complete binary tree that has n leaf nodes, where n is the size of the current group. Fig 2 depicts the logical binary tree containing seven members. Each member is associated with a single node and this node will be called the member node. Here, $H : \{0,1\}^* \rightarrow \{0,1\}^{|K|}$ denotes a collision-resistant hash function, where $|K|$ is the length of the key, $||$ denotes bitwise concatenation, and $\oplus$ denotes XOR operation. One key, denoted as K_X, and two blind keys, denoted as L_X and R_X, are associated with each internal nodes. GKM computes all the blind keys in the following way.
  - It first randomly selects the node key. The cost of randomly selecting the key can be reduced by computing the key $K_X = H(c||K_L||K_R)$, where K_L and K_R are the left and right child node key, respectively.
  - Blind keys of the node are computed by XORing the node key with a hash value computed using the child node key and a counter c. See Fig 2 for the actual equation used to compute the blind keys.
- **Step 2.** GKM multicasts all the blind keys with the tree information in a single message to all members.
- **Step 3.** Each user retrieves blind keys related to each user from the message. Each user requires a single blind key from each node on its key path. Then each user computes the group key. For example, user 1 in Fig 2 computes the group key K_G using the following equations:

$$K_{12} = L_{12} \oplus H(c||K_1),$$
$$K_{14} = L_{14} \oplus H(c||K_{12}),$$
$$K_G = L_G \oplus H(c||K_{14}).$$

  Therefore, a user require maximum d XOR and d hash operations to compute the group key, where d is the height of the tree.

3.2 The Join Protocol

The join and leave protocol are both similar to TGDH protocol of Kim et al.'s [5]. When a member wants to join the group, it contacts the GKM and authenticates itself to it. If the GKM decides to allow the user to join the group using some pre-established policy, it establishes a symmetric key with the user. The protocol used to establish the symmetric key is omitted in this paper. It then use the following protocol to update the group key.

- **Step 1.** It determines the insertion node in the tree. The insertion node is the shallowest rightmost node, where join does the increase the height of the tree. If the tree is a full binary tree, the new member joins at the root to reduce computation and communications cost. This is shown in Fig 3.

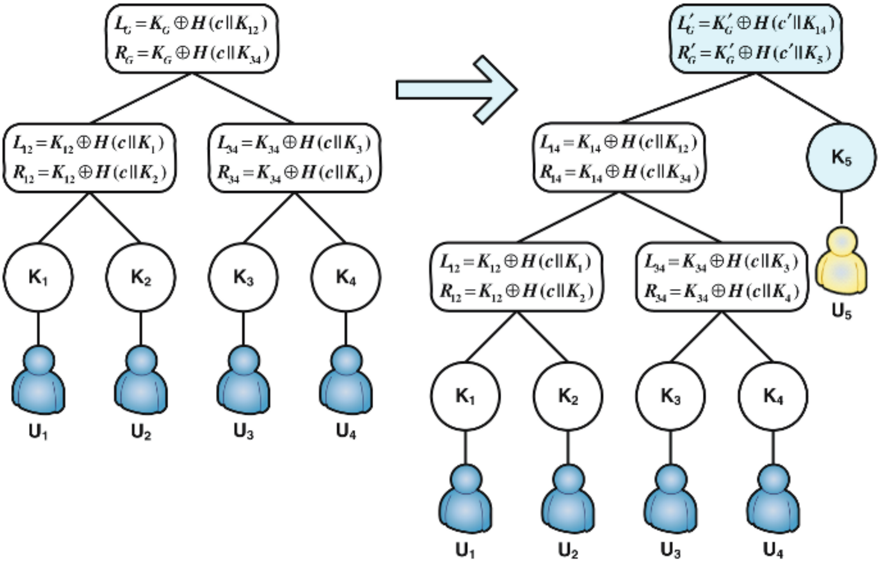

$L_G = K_G \oplus H(c||K_{12})$
$R_G = K_G \oplus H(c||K_{34})$
$L_{12} = K_{12} \oplus H(c||K_1)$
$R_{12} = K_{12} \oplus H(c||K_2)$
$L_{34} = K_{34} \oplus H(c||K_3)$
$R_{34} = K_{34} \oplus H(c||K_4)$
K1
K2
K3
K4
U1
U2
U3
U4
$L'_G = K'_G \oplus H(c'||K_{14})$
$R'_G = K'_G \oplus H(c'||K_5)$
$L_{14} = K_{14} \oplus H(c||K_{12})$
$R_{14} = K_{14} \oplus H(c||K_{34})$
K5
U5
$L_{12} = K_{12} \oplus H(c||K_1)$
$R_{12} = K_{12} \oplus H(c||K_2)$
$L_{34} = K_{34} \oplus H(c||K_3)$
$R_{34} = K_{34} \oplus H(c||K_4)$
K1
K2
K3
K4
U1
U2
U3
U4

Fig. 3. Tree Updating in Join Operation. K_{14} in the right tree is equal to K_G in the left tree.

- **Step 2.** The GKM updates all the keys and blind keys of nodes on the key path of the insertion node. The counter c is increased and is used to compute the blind keys. This is required to preserve backward secrecy. In Fig 3, although the new user can compute $H(c'||K_{14})$, it cannot obtain K_{14}, which is the previous group key, from it.
- **Step 3.** The GKM multicasts all the new blind keys in a single messages to all the members including the newly joined member.

3.3 The Leave Protocol

When a member decides to leave or is exiled from the group, the GKM uses the following protocol to update the group key.

- **Step 1.** The GKM promotes the sibling node of the leaving member node to the parent node and recalculates the parent of the promoted node with a incremented counter.
- **Step 2.** The GKM updates node keys and the corresponding blind keys of nodes on the key path from the parent of the promoted node.
- **Step 3.** The GKM multicasts all the new blind keys in a single message to the remaining members.

Fig 4 represents the key tree when user 3 leaves the group from the key tree depicted in Fig 2. Since the user 3 does not know K_{12}, K_4, or K_{57}, he/she cannot compute the new group key using the values he/she already knows and the blind keys transmitted during the key update. Therefore, our scheme preserves forward secrecy.

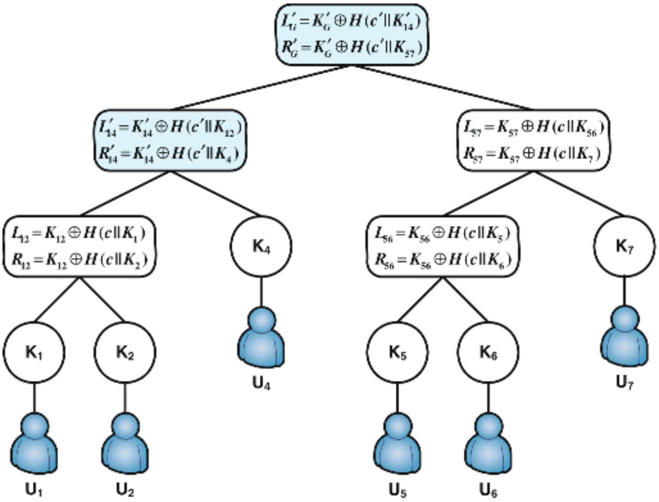

$L'_G = K'_G \oplus H(c' \| K'_{14})$
$R'_G = K'_G \oplus H(c' \| K_{57})$
$L'_{14} = K'_{14} \oplus H(c' \| K_{12})$
$R'_{14} = K'_{14} \oplus H(c' \| K_4)$
$L_{57} = K_{57} \oplus H(c \| K_{56})$
$R_{57} = K_{57} \oplus H(c \| K_7)$
$L_{12} = K_{12} \oplus H(c \| K_1)$
$R_{12} = K_{12} \oplus H(c \| K_2)$
K_4
$L_{56} = K_{56} \oplus H(c \| K_5)$
$R_{56} = K_{56} \oplus H(c \| K_6)$
K_7
K_1
K_2
U_4
K_5
K_6
U_7
U_1
U_2
U_5
U_6

Fig. 4. Tree Updating in Leave Operation. This tree is a result of user 3 leaving the group from the tree given in Fig 2.

4 Analysis

4.1 Security Analysis

In this subsection, we will prove that our scheme preserves forward and backward secrecy requirement.

Theorem 1. *If Bresson et al.'s scheme is secure and preserves forward and backward secrecy, our scheme is also secure and preserves forward and backward secrecy requirement.*

Proof. Bresson et al. have proven that their protocol is secure using the random oracle model [6]. If we take a subtree of height 1 containing two leaf nodes from our tree, the protocol is identical to Bresson et al.'s scheme. From this point of view, the same proof can be applied to our system when the tree height is 1 with total three nodes. We will regard this as the basis of the induction in this proof. Let's assume that trees of height $d-1$ are secure and preserves forward and backward secrecy. Now, let's consider a tree of height d. By induction hypothesis, the keys located at the root of the left/right subtree are secure and preserve forward and backward secrecy. Moreover, a tree of height d can be regarded as a tree of height 1 by considering the left/right subtree of the root as a single node. This is identical to the base case. As a result, our scheme is secure and preserves forward and backward secrecy requirement. □

4.2 Efficiency Analysis

We will first compare our scheme with other schemes with respect to communication cost. During the key update, Bresson et al.'s scheme requires n unicasts each consisting

Table 1. The Worst Case Comparison of the Communication Cost

Protocol	Join		Leave
	Multicast	Unicast	
LKH	$2d\|K\|$	0	$2(d-1)\|K\|$
OFT	$d\|K\|$	$d\|K\|$	$(d-1)\|K\|$
Bresson et al.	$n\|K\|$	0	$n\|K\|$
Our Scheme	$2d\|K\|$	0	$2(d-1)\|K\|$

Except for the OFT scheme, we assume that key update information are sent in a single multicast message to all the members including the newly joined member.
d: the height of the tree after join or leave, n: the number of group members after join or leave, $|K|$: the size of the key,
E: encryption operation, D: decryption operation, X: XOR operation, H: hash operation

Table 2. The Worst Case Comparison of the Computation Cost

Protocol	Join		Leave	
	GKM	Member	GKM	Member
LKH	$2dE$	dD	$2(d-1)E$	$(d-1)D$
OFT	$(d+1)E+2dH$	$1D+dH$	$(d-1)(E+H)$	$1D+(d-1)H$
Bresson et al.	$nX+(n+1)H$	$1X+1H$	$nX+(n+1)H$	$1X+1H$
Our Scheme	$d(2X+3H)^{\dagger}$	$d(X+H)$	$(d-1)(2X+3H)$	$(d-1)(X+H)$

We did not include the cost of authenticating a user during the join process. The notations used in this table is described in Table 1.
† We have included the hash operation required to compute the node key. LKH scheme, however, also requires some operation to compute the node key which is omitted here.

of a single blind key. Since these data needs no protection, we can also send n blind keys in a single multicast message to all the members. We will compare our scheme with this version of Bresson et al.'s. Our scheme sends all the newly updated blind keys in a single multicast message to all the members including the newly joined member. As shown in Table 1, in our scheme, the size of this multicast message is in order of $O(\log n)$, where as Bresson et al.'s scheme is $O(n)$. However, this comparison is based on worst-case analysis. Therefore, on the average, ours will save the network bandwidth significantly compared to Bresson et al.'s. With respect to communication cost, our scheme is equal to LKH, while OFT is less than ours.

Now, we will compare our scheme with others with respect to computational cost. Since our scheme do not require any symmetric or public key operations, it is clear that our system outperforms previous centralized schemes such as LKH and OFT. This is shown in Table 2. Both ours and Bresson et al.'s scheme only use XOR and hash operations. The computational cost of GKM is in order of $O(\log n)$, whereas Bresson et al.'s scheme is $O(n)$. However, with respect to computational cost of a member, Bresson et al.'s scheme is $O(1)$, whereas our scheme requires maximum $O(d)$.

4.3 Additional Benefits of Our Scheme Compared to Others

There are following additional benefits of our scheme compared to other schemes.

- First, since our scheme is hierarchical scheme with keys associated with each internal node, these keys can be used as a subgroup key. Bresson et al.'s scheme does not provide this function.
- Second, in OFT scheme, one of the members must change their shared key with GKM when updating the key during the leave protocol. However, our scheme only requires incrementing the counter instead of changing the shared key.

5 Conclusion

In this paper, we proposed a new centralized hierarchical group key management protocol suitable for a ubiquitous computing environment. The proposed scheme is very practical, since the protocol only use XOR and hash operations during group key updates. It is also scalable, since the order of total message size sent during key updates is $O(\log n)$, where n is the size of the group. Moreover, there are some additional benefits of our scheme, which is enumerated in the previous section, compared to others. We have also proved the security of our proposed protocol.

References

1. Rafaeli, S., Hutchison, D.: A Survey of Key Management for Secure Group Communication. ACM Computing Surveys, Vol. 35, No. 3. (2003) 309–329
2. Wong, C.K., Gouda, M.G., Lam, S.S.: Secure Group Communications Using Key Graphs. IEEE/ACM Trans. on Netw., Vol. 8, No. 1. (2000) 16–30
3. McGrew, D.A., Sherman, A.T.: Key Establishment in Large Dynamic Groups using One-way Function Trees. TIS Labs at Network Associates, Tech. Rep. No. 0755. (1998)
4. Mittra, S.: Iolus: A Framework for Scalable Secure Multicasting. In Proc. of the 6th ACM Conf. on Computer and Communications Security. (1999) 101–112
5. Kim, Y., Perrig, A., Tsudik, G.: Simple and Fault-Tolerant Key Agreement for Dynamic Collaborative Groups. In Proc. of the 7th ACM Conf. on Computer and Communications Security. (2003) 235–244
6. Bresson, E., Chevassut, O., Essiari, A., Pointcheval, D.: Mutual Authentication and Group Key Agreement for Low-Power Mobile Devices. J. of Computer Communications, Vol. 27. No. 17. Elsevier (2004) 1730–1737

Efficient User Authentication and Key Agreement in Ubiquitous Computing

Wen-Shenq Juang

Department of Information Management,
Shih Hsin University,
No. 1, Lane 17, Sec. 1, Muja Rd., Wenshan Chiu,
Taipei, Taiwan, 116, R.O.C
wsjuang@cc.shu.edu.tw

Abstract. In ubiquitous computing, many computers serve each person at any time and any place. These computers could be thin servers and only have low computation and communication capacity. In this paper, we propose a novel user authentication and key agreement scheme suitable for ubiquitous computing environments. The main merits include: (1) there are many security domains which have their own security controllers, and each security domain can be formed dynamically; (2) a user only has to register in a security controller once, and can use all permitted services in this environment; (3) a user can freely choose his own password to protect his secret token; (4) the computation and communication cost is very low; (5) servers and users can authenticate each other; (6) it generates a session key agreed by the server and the user; (7) our proposed scheme is a nonce-based scheme which does not have a serious time-synchronization problem.

Keywords: User Authentication, Session Key, Ubiquitous Computing, Smart Card, Network Security.

1 Introduction

In ubiquitous computing, a user may use many computers at any time and any place, while he does not need to know how to use these computers [2, 21]. These computers could be thin servers and only have low computation and communication capacity. When a user enters his home, corporation, or another unfamiliar place, if this user wants to use the permitted services provided by the connected servers, he must pass the authentication of these servers.

In 1981, Lamport [13] proposed a password authentication scheme for verifying the validity of users. Since then, many schemes have been proposed to point out its drawbacks and improve the efficiency and security of Lamport's scheme [5, 10, 11, 12, 19, 23]. Only passing a password for authenticating between the user and the server is not enough, since the password is easily tapped by the adversary. Before two parties can do secure communication, a session key must be exchanged for protecting subsequence communications [1, 7, 8, 18, 22]. Also,

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 396–405, 2006.

using smart cards [5, 7, 8, 11, 19, 23], user authentication and key agreement can be simplified, efficient and flexible for creating a secure distributed computers environment.

For basic security and efficient requirements, the following criteria are important for user authentication and key agreement schemes in ubiquitous computing environments [2, 7, 8, 21].

C1: Dynamic participation: A security domain can be formed dynamically and a user can join a security domain at any time.
C2: Single registration: A user only needs to register in a single security controller and can use all permitted services in dynamically joining servers.
C3: Freely chosen password: A user can freely choose and change his password for protecting his secret token, e.g. a smart card.
C4: Low computation and communication cost: Since capacity and communication constrains of mobile devices or thin servers, they may not offer a powerful computation capability and high bandwidth.
C5: Mutual authentication: A user and a server can authenticate each other.
C6: Session key agreement: A user and a server must negotiate a session key for subsequent communications.

In this paper, we propose an efficient user authentication and key agreement scheme for ubiquitous computing environments. Our scheme is very efficient since our scheme only uses the symmetric cryptosystems and hashing functions. Our proposed scheme satisfies all above six criteria. Also, our proposed scheme has no serious time-synchronization problem since our scheme is based on nonces.

The remainder of this paper is organized as follows: In Section 2, we describe a high-level system architecture for our proposed ubiquitous computing environment. In Section 3, we present our user authentication and key agreement scheme suitable for ubiquitous computing. In Section 4, the security analysis for our proposed scheme is given. The performance consideration for our proposed scheme is given in Section 5. In Section 6, we make a discussion. Finally, a concluding remark is given in Section 7.

2 System Architecture

In this section, we describe a general high-level system architecture for our proposed ubiquitous computing environment. In this architecture, some computation nodes in a nearby region will form a security domain which contains a security controller and many member nodes. A subscriber or user can register in one or more security domains and do communications with other nodes via wireline or wireless communications. Any two security controllers can share a secret key via current well-used key distribution schemes. Any user who plans to acquire a service has to register himself with at least one security controller and becomes a subscriber to this security domain. The security domain that a user registered with is referred to as his home security domain, and other security domain that the user visits are his visiting security domains. A user can do communication directly with other available users in the same visiting security domain. If a subscriber want to communicate with another user in another security domain, a suitable routed protocol must be provided. For simplicity, we only assume a user is registered in a security domain not in multiple security domains.

3 User Authentication and Key Agreement in Ubiquitous Computing

There are four kinds of participants for a particular user in our user authentication and key agreement protocol: a user, a server, the home security controller of the user, and the home security controller of the server. Let $U_{i,j}$ denote user j registered in security controller i, SD_i denote the security domain i and SC_i denote the security controller i. So SD_i is the home security domain of $U_{i,j}$ and $SD_{j,j\neq i}$ are visiting security domains of $U_{i,j}$. Let $ID_{U_{i,j}}$ be a unique identification of $U_{i,j}$ and ID_{SC_i} be a unique identification of the security controller SC_i. Let "$X \rightarrow Y : Z$" denote that a sender X sends a message Z to a receiver Y, $E_k(m)$ denote the ciphertext of m encrypted using the secret key k of some secure symmetric cryptosystem [16], $D_k(c)$ denote the plaintext of c decrypted using the secret key k of the corresponding symmetric cryptosystem [16], "$||$" denote the conventional string concatenation operator and $\oplus$ denote the bitwise exclusive-or operator. Let h be a public one-way function [17]. Let x_i be the master secret key kept secretly by the security controller SC_i. Also let $\eta_{i,k}$ be the shared secret key between two security controllers SC_i and $SC_{k.}$ Our proposed protocol is as follows:

Home Security Controller Registration Phase: Assume $U_{i,j}$ submits his identity $ID_{U_{i,j}}$ and his password $PW_{U_{i,j}}$ to his home security controller SC_i for registration. If SC_i accepts this request, he will perform the following steps:
Step 1: Compute $U_{i,j}$'s secret information $\alpha_{i,j} = h(x_i, ID_{U_{i,j}})$ and $\beta_{i,j} = \alpha_{i,j} \oplus PW_{U_{i,j}}$.
Step 2: Store $ID_{U_{i,j}}$, and $\beta_{i,j}$ to the memory of a smart card and issue this smart card to $U_{i,j}$ or send them secretly to $U_{i,j}$.
Shared Key Inquiry Phase: If $U_{i,j}$ wants to use the services provided by $U_{k,l}$, these two users must share a secret key $\lambda_{i,j,k,l}$ for user authentication and key agreement. $U_{i,j}$ can compute the shared secret key $\lambda_{i,j,k,l}$ from $\alpha_{i,j}$, ID_{SC_k} and $ID_{U_{k,l}}$ when he does user authentication and key agreement. If $U_{k,l}$ has not the shared secret key $\lambda_{i,j,k,l}$, he must query it from his home security controller SC_k and SC_k will forward this query to SC_i. SC_i will compute $\gamma_{i,j,k} = h(\alpha_{i,j}||ID_{SC_k})$, and then sends $\gamma_{i,j,k}$ to SC_k. Then SC_k can compute $\lambda_{i,j,k,l} = h(\gamma_{i,j,k}||ID_{U_{k,l}})$ and send $\lambda_{i,j,k,l}$ to $U_{k,l}$. In this phase, they will perform the following steps:
Step 1: $U_{k,l} \rightarrow SC_k : N_1, ID_{U_{i,j}}, ID_{U_{k,l}}$
Step 2: $SC_k \rightarrow SC_i : N_2, ID_{SC_k}, E_{\eta_{i,k}}(ID_{U_{i,j}}, KR, h(ID_{U_{i,j}}||ID_{SC_k} ||KR||N_2))$
Step 3: $SC_i \rightarrow SC_k : E_{\eta_{i,k}}(\gamma_{i,j,k}, h(ID_{U_{i,j}}||ID_{SC_k}||KR||N_2||\gamma_{i,j,k}))$
Step 4: $SC_k \rightarrow U_{k,l} : E_{\alpha_{k,l}}(\lambda_{i,j,k,l}, h(ID_{U_{i,j}}||ID_{SC_k}||ID_{U_{k,l}}||KR||N_1|| \lambda_{i,j,k,l}))$

In Step 1, $U_{k,l}$ sends a nonce N_1, the identifications $ID_{U_{i,j}}, ID_{U_{k,l}}$ to his home security controller SC_k, where N_1 is for freshness checking.

Upon receiving the message in Step 1, SC_k first checks if $\gamma_{i,j,k}$ is in his shared keys table. If not, he sends a nonce N_2, his identification ID_{SC_k} and the encrypted message $E_{\eta_{i,k}}(ID_{U_{i,j}}, KR, h(ID_{U_{i,j}}||ID_{SC_k} ||KR||N_2))$ to SC_i, where KR is the key request message. If yes, goes to Step 4.

Upon receiving the message in Step 2, SC_i decrypts the message $E_{\eta_{i,k}}(ID_{U_{i,j}}, KR, h(ID_{U_{i,j}}||ID_{SC_k} ||KR||N_2))$, and checks if the verification tag $h(ID_{U_{i,j}} ||ID_{SC_k} ||KR||N_2)$ is valid and the nonce N_2 is fresh. If yes, he computes $\gamma_{i,j,k} = h(\alpha_{i,j}||ID_{SC_k})$ and then sends the encrypted message $E_{\eta_{i,k}}(\gamma_{i,j,k}, h(ID_{U_{i,j}} ||ID_{SC_k}||KR|| N_2||\gamma_{i,j,k}))$ back to SC_k. Since the nonce N_2 is not chosen by SC_i, for checking the freshness of the nonce N_2 in practical implementation, SC_i can keep a recently used nonces table for each security controller. Since this phase only does shared keys inquiry, the replay of the older message only causes SC_i to resent an additional encrypted message back to SC_k.

Upon receiving the message in Step 3, SC_k decrypts the message $E_{\eta_{i,k}}(\gamma_{i,j,k}, h (ID_{U_{i,j}}||ID_{SC_k}||KR|| N_2||\gamma_{i,j,k}))$ and checks if the nonce N_2 is fresh and the verification tag $h(ID_{U_{i,j}}|| ID_{SC_k} ||KR||N_1||\gamma_{i,j,k})$ is valid. If yes, he records $(ID_{U_{i,j}}, \gamma_{i,j,k})$ in a key table, computes $\lambda_{i,j,k,l} = h(\gamma_{i,j,k}||ID_{U_{k,l}})$ and then sends the encrypted message $E_{\alpha_{k,l}}(\lambda_{i,j,k,l}, h(ID_{U_{i,j}}||ID_{SC_k}||ID_{U_{k,l}}||KR||N_1|| \lambda_{i,j,k,l}))$ back to $U_{k,l}$.

Upon receiving the message in Step 4, $U_{k,l}$ computes $\alpha_{k,l} = \beta_{k,l} \oplus PW_{U_{k,l}}$ and decrypts the message $E_{\alpha_{k,l}}(\lambda_{i,j,k,l}, h(ID_{U_{i,j}}||ID_{SC_k}||ID_{U_{k,l}}||KR||N_2|| \lambda_{i,j,k,l}))$ and checks if the nonce N_2 is fresh and the verification tag $h(ID_{U_{i,j}}||ID_{SC_k}|| ID_{U_{k,l}}||KR||N_1|| \lambda_{i,j,k,l})$ is valid. If yes, he records $(ID_{U_{i,j}}, \lambda_{i,j,k,l})$ in a shared keys table.

User Authentication and Session Key Agreement Phase: If $U_{i,j}$ wants to authenticate $U_{k,l}$ and agree a session key sk_n for nth session, he must attach his smart card to a card reader. He then inputs his identity $ID_{U_{i,j}}$ and his password $PW_{U_{i,j}}$ to this device. The following protocol is the nth user authentication and key agreement for $U_{i,j}$ with respect to $U_{k,l}$.

Step 1: $U_{i,j} \rightarrow U_{k,l} : N_3, ID_{U_{i,j}}, E_{\lambda_{i,j,k,l}}(ru_n, h(N_3||ID_{U_{i,j}}||ru_n))$
Step 2: $U_{k,l} \rightarrow U_{i,j} : N_4, E_{\lambda_{i,j,k,l}}(rs_n, h(N_3||N_4||ID_{U_{k,l}}||rs_n))$
Step 3: $U_{i,j} \rightarrow U_{k,l} : E_{sk_n}(N_4 + 1)$

In step 1, $U_{i,j}^{\prime}s$ smart card first computes $\alpha_{i,j} = \beta_{i,j} \oplus PW_{U_{i,j}}$, $\gamma_{i,j,k} = h(\alpha_{i,j}||ID_{SC_k})$ and $\lambda_{i,j,k,l} = h(\gamma_{i,j,k}||ID_{U_{k,l}})$ and sends his identification $ID_{U_{i,j}}$, a nonce N_3 and the encrypted message $E_{\lambda_{i,j,k,l}}(ru_n, h(N_3||ID_{U_{i,j}}||ru_n))$ to $U_{k,l}$. The nonce N_3 is for freshness checking. The encrypted message includes the nth random value ru_n, which is used for generating the nth session key sk_n, and the authentication tag $h(N_3||ID_{U_{i,j}}||ru_n)$, which is for verifying the identification of $U_{i,j}$.

Upon receiving the message in step 1, $U_{k,l}$ first checks if $\lambda_{i,j,k,l}$ is in his shared keys table. If not, he does shared key inquiry to find it. He then decrypts the message $E_{\lambda_{i,j,k,l}}(ru_n, h(N_3||ID_{U_{i,j}}||ru_n))$ and verifies if the authentication tag $h(N_3||ID_{U_{i,j}}||ru_n)$ is valid by using the shared key $\lambda_{i,j,k,l}$. If it is valid, $U_{k,l}$ sends a nonce N_4 and the encrypted message $E_{\lambda_{i,j,k,l}}(rs_n, h(N_3||N_4||ID_{U_{k,l}}||rs_n))$ back to $U_{i,j}$. The encrypted message includes the random value rs_n chosen by $U_{k,l}$, which is used for generating the *nth* session key $sk_n = h(ru_n, rs_n, \lambda_{i,j,k,l})$, and the nonce N_4, which is for freshness checking.

Upon receiving the message in step 2, $U_{i,j}$ decrypts the message by computing $D_{\lambda_{i,j,k,l}}(E_{\lambda_{i,j,k,l}}(rs_n, h(N_3||N_4||ID_{U_{k,l}}||rs_n)))$. He then checks if the authentication tag $h(N_3||N_4||ID_{U_{k,l}}||rs_n)$ is in it for freshness checking. If yes, $U_{i,j}$ computes the session key $sk_n = h(ru_n, rs_n, \lambda_{i,j,k,l})$ and sends the encrypted message $E_{sk_n}(N_4 + 1)$ back to $U_{k,l}$.

After receiving the message in step 3, $U_{k,l}$ decrypts the message by computing $D_{sk_n}(E_{sk_n}(\ N_4+1))$ and checks if the nonce N_4+1 is in it for freshness checking. Then $U_{i,j}$ and $U_{k,l}$ can use the session key $sk_n = h(ru_n, rs_n, \lambda_{i,j,k,l})$ in secure communication soon

4 Security Analysis

(1) Mutual authentication [3]

Let $X \overset{K}{\leftrightarrow} Y$ denotes the player X shares a session key K with the player Y. Thus mutual authentication is complete between $U_{i,j}$ and $U_{k,l}$ if there is a session key sk_n such that $U_{i,j}$ believes $U_{i,j} \overset{sk_n}{\leftrightarrow} U_{k,l}$, and $U_{k,l}$ believes $U_{i,j} \overset{sk_n}{\leftrightarrow} U_{k,l}$ for the *nth* transaction [3]. A strong mutual authentication may add the following statement: $U_{i,j}$ believes $U_{k,l}$ believes $U_{i,j} \overset{sk_n}{\leftrightarrow} U_{k,l}$, and $U_{k,l}$ believes $U_{i,j}$ believes $U_{i,j} \overset{sk_n}{\leftrightarrow} U_{k,l}$ for the *nth* transaction.

In step 1 of the user authentication and session key agreement phase, after $U_{k,l}$ receives the message $E_{\lambda_{i,j,k,l}}(ru_n, h(N_3||ID_{U_{i,j}}||ru_n))$, $U_{k,l}$ will compute $D_{\lambda_{i,j,k,l}}(E_{\lambda_{i,j,k,l}}(ru_n, h(N_3||ID_{U_{i,j}}||ru_n)))$ using the shared key $\lambda_{i,j,k,l}$ of $U_{i,j}$ and $U_{k,l}$. Then $U_{k,l}$ can check if this authenticator $h(N_3||ID_{U_{i,j}}||ru_n)$ is valid. If yes, $U_{k,l}$ chooses a random number rs_n and can computes the *nth* session key $sk_n = h(ru_n, rs_n, \lambda_{i,j,k,l})$ and believes $U_{i,j} \xleftrightarrow{sk_n} U_{k,l}$. In step 2 of the user authentication and session key agreement phase, upon receiving the message $E_{\lambda_{i,j,k,l}}(rs_n, h(N_3||N_4||ID_{U_{k,l}}||rs_n))$, $U_{i,j}$ decrypts the message $D_{\lambda_{i,j,k,l}}(E_{\lambda_{i,j,k,l}}(rs_n, h(N_3||N_4||ID_{U_{k,l}}||rs_n)))$ and confirms if this message contains the authenticator $h(N_3||N_4||ID_{U_{k,l}}||rs_n)$. If yes, $U_{i,j}$ generates a session key $sk_n = h(ru_n, rs_n, \lambda_{i,j,k,l})$ and believe $U_{i,j} \xleftrightarrow{sk_n} U_{k,l}$. Since N_3 is chosen by $U_{i,j}$, $U_{i,j}$ will believes $U_{k,l}$ believes $U_{i,j} \xleftrightarrow{sk_n} U_{k,l}$. In step 3 of the user authentication and session key agreement phase, after $U_{k,l}$ receiving $E_{sk_n}(N_4 + 1)$, he will decrypt this message $E_{sk_n}(N_4 + 1)$ with the *nth* session key sk_n and get $N_4 + 1$. Then $U_{k,l}$ checks if N_4 which is sent by him is correct. If yes, $U_{k,l}$ believes $U_{i,j}$ believes $U_{i,j} \xleftrightarrow{sk_n} U_{k,l}$.

(2) Session key agreement

The session key $sk_n = h(ru_n, rs_n, \lambda_{i,j,k,l})$ is known to nobody but $U_{i,j}$ and $U_{k,l}$, since the random values ru_n, rs_n are randomly chosen by $U_{i,j}$ and $U_{k,l}$ and are encrypted by the shared key $\lambda_{i,j,k,l}$.

(3) Withstanding attacks
We prove our user authentication and key agreement scheme can resist to the following attacks.

1. The man-in-middle attack [18]
Our proposed scheme can resist to the man-in-the-middle attack. If the message is modified by the adversary, either ends of the communication will find out and reject this message. Since our proposed scheme can accomplish strong mutual authentication, our scheme can resist this attack.
2. The dictionary attack [1]
For deriving the session key sk_n, the adversary must know ru_n, rs_n and $\lambda_{i,j,k,l}$ but the shared key $\lambda_{i,j,k,l}$ is only kept secretly by $U_{i,j}$ and $U_{k,l}$, the security controllers SC_i and SC_k. The adversary can not get the session key sk_n, since ru_n and rs_n are randomly chosen and protected by the shared key $\lambda_{i,j,k,l}$ and the entropy of ru_n, rs_n or $\lambda_{i,j,k,l}$ is very large.
3. The replay attack [20]
The replay attack is replaying the message to the user or the server. Our proposed scheme also provide an ability to avoid this attack. Our proposed scheme in the user authentication and key agreement phase uses the nonces N_3, N_4 to resist the replay attack. In the shared key inquiry phase, the nonces N_1 and N_2 is used for $U_{k,l}$ and SC_k to resist the replay attack. Also, in the shared key inquiry phase, SC_i only does the response of shared keys inquiry, the replay of the older message only causes SC_i to resent an additional encrypted message back to SC_k. Since the nonce N_2 is not chosen by SC_i, for checking the freshness of the nonce N_2 in practical implementation, SC_i can keep a recently used nonces table.
4. The modification attack [24]
Upon receiving the message $N_3, ID_{U_{i,j}}, E_{\lambda_{i,j,k,l}}(ru_n, h(N_3||ID_{U_{i,j}}||ru_n))$ in step 1 of the user authentication and session key agreement phase, the adversary can not alter this message since the adversary does not has the shared key $\lambda_{i,j,k,l}$. If the adversary modifies the message, the server will reject this message. In the other hand, the user also can observe the original message whether it is changed by the adversary. So this attack on our scheme can be prevented.
5. The insider attack [10]
The weak password $PW_{U_{i,j}}$ used in our scheme is only for protecting the corresponding smart card from being used by illegal users. If a user uses $PW_{U_{i,j}}$ to register in several security controllers for his convenience, the insider of the security controller can not impersonate the user to access other servers in another security controllers if these security controllers do not have the corresponding smart card. If the authentication scheme used in other servers is not the same as our proposed scheme, the password derived by the insider attack may work. In this situation, we can replace $\beta_{i,j} = \alpha_{i,j} \oplus PW_{U_{i,j}}$ with $\alpha_{i,j} \oplus h(b \oplus PW_{U_{i,j}})$ and use the checking method mentioned in [10] for protecting the weak password being known by SC_i.

Table 1. Efficiency comparison between our scheme and other related schemes for user authentication and session key agreement phase

	Our scheme	Juang's scheme [7]	Juang's scheme [8]
E1	160 bits	160 bits	160 bits
E2	512 bits	512 bits	512 bits
E3	1 Hash	1 Hash	1 Hash
E4	8 Sym + 8 Hash	None	4 Sym +2 Hash
E5	6 Sym + 6 Hash	4 Sym + 3 Hash	5 Sym + 3 Hash

E1: Memory needed in the smart card for cryptographic parameters
E2: Communication cost of the authentication for cryptographic parameters
E3: Computation cost of the registration
E4: Computation cost of the shared key inquiring
E5: Computation cost of the user authentication and key agreement
Hash: Hashing operation Exp: Exponential operation
Sym: Symmetric encryption or decryption

5 Performance Considerations

In this section, we present an efficiency comparison among our proposed scheme and related schemes [7, 8]. The comparison is given in Table 1. We assume the output size of secure one-way hashing functions [17] is 160 bits and the block size of secure symmetric cryptosystems [16] is 128 bits. In our scheme, the memory needed in the smart card for storing cryptographic parameters $\beta_{i,j}$ is 160 bits. In our proposed scheme, the communication cost of the user authentication and key agreement for cryptographic parameters $E_{\lambda_{i,j,k,l}}(ru_n, h(N_3||ID_{U_{i,j}}||ru_n))$ and $E_{\lambda_{i,j,k,l}}(rs_n,h(N_3||N_4||ID_{U_{k,l}}||rs_n))$ is (96+160)+(96+160)=512 bits, where ru_n and rs_n can both be of 96 bits. For implicit mutual authentication [22], Step 3 can be delayed to the subsequent private communication. The computation cost of registration is 1 hash operation and 1 exclusive-or operation. The computation cost of the shared key inquiry phase is 4 symmetric key encryptions, 4 symmetric key decryptions, and 8 hash operations. The computation cost of user authentication and key agreement in our scheme is 3 symmetric key encryptions, 3 symmetric key decryptions and 6 hash operations and 1 exclusive-or operation.

We summarize the functionality and complexity of our scheme and related schemes in Table 2. Our scheme can satisfy all listed functions and has low communication and computation cost. In comparison with Juang's schemes [7, 8], our scheme provides an ability of dynamic participation which is not provided by Juang's schemes [7, 8].

6 Discussions

Only one-way hashing functions and symmetric cryptosystems are used in our proposed scheme. Our approach provides another choice for better efficiency and no need to base on any assumed hard number theoretical problem, e.g.,

Table 2. Functionality comparison between our scheme and other related schemes

	Our scheme	Juang's scheme [7]	Juang's scheme [8]
C1	Yes	No	No
C2	Yes	No	Yes
C3	Yes	Yes	Yes
C4	Very low	Very low	Very low
C5	Yes	Yes	Yes
C6	Yes	Yes	Yes
C7	Yes	Yes	Yes

C1: Dynamic participation
C2: Single registration
C3: Freely chosen password
C4: Communication and computation cost
C5: Mutual authentication
C6: Session key agreement
C7: No serious time synchronization problem

the discrete logarithm problem or the factoring problem [14]. In practical considerations, a one-way hash function can be easily constructed by a symmetric cryptosystem [15]. This approach can reduce the memory in a smart card for storing cryptographic programs.

In our scheme, for improving the repairability mentioned in [6, 10], the secret value $\alpha_{i,j} = h(x_i, ID_{U_{i,j}})$ stored in each $U_{i,j}$'s smart card can be replaced with the new formula $\alpha_{i,j} = h(x_i, ID_{U_{i,j}}, t)$, where t is the number of times that $U_{i,j}$ has revoked his used secret key $\alpha_{i,j}$. But this approach will need SC_i to record the number t in his database.

The password changing procedure proposed in [10] can be directly used in our proposed scheme for changing users' passwords. In ubiquitous computing environments [2, 21], users do not need to know how to use his computers for convenience. The password for protecting the secure token in our scheme can be disable in order to make users friendly to use his computers.

Like the schemes in [7, 8], we do not provide the perfect forward secrecy in our proposed scheme, since it may cause a result of lower performance and increased communication and computation cost. If this property is required, the Diffie-Hellman algorithm [4] can be directly used in our scheme as in the schemes [7, 8].

For managing the shared keys between many security controllers, some security controllers can be a group and form a security domain, the key generation and distribution between these security controllers can be the same with our proposed scheme.

In [9], Juang used the concept of dynamic pseudo identifications for protecting user's identification without using public-key cryptosystems. This concept can also be applied to our proposed scheme for providing the anonymity of users. We are current working designing an enhanced privacy protection user authentication and key agreement scheme in ubiquitous computing environments using the concept proposed in [9].

7 Conclusions

In this paper, we have proposed an efficient user authentication and key agreement scheme in ubiquitous computing environments. In our scheme, a security domain can be formed dynamically and a user can join a security domain at any time. Also, a user only needs to register one time in a security controller and can use his secret key to generate all shared secret keys with all other eligible users in the connected world. Our scheme also has low communication and computation cost for user authentication and key agreement by only using symmetric cryptosystems and one-way functions. Also, our scheme successfully solves the serious time-synchronization problem in a distributed computers environment since our proposed scheme is nonce-based.

Acknowledgment. This work was supported in part by the National Science Council of the Republic of China under the Grant NSC-94-2213-E-128-001, and by the Taiwan Information Security Center (TWISC), National Science Council under the Grants NSC 94-3114-P-001-001-Y and NSC 94-3114-P-011-001.

References

1. S. Bellovin and M. Merritt, "Encrypted key Exchange: Password-Based Protocols Secure Against Dictionary Attacks," Proceedings of IEEE Symposium on Research in Security and Privacy, pp. 72-84, 1992.
2. G. Borriello, "Key Challenges in Communication for Ubiquitous Computing," *IEEE Communications Magazine*, pp. 16-18, May 2002.
3. M. Burrows, M. Abadi and R. Needham, "A Logic of Authentication," *ACM Trans. on Computer Systems*, Vol. 8, No. 1, pp. 18-36, 1990.
4. W. Diffie and M. Hellman, "New Directions in Cryptography," *IEEE Transactions on Information Theory*, Vol. IT-22, No. 6, pp. 644-654, 1976.
5. C. Fan, Y. Chan and Z. Zhang, "Robust Remote Authentication Scheme with Smart Cards," *Computers & Security*, Vol. 24, pp. 619-628, 2005.
6. T. Hwang and W. Ku, "Repairable Key Distribution Protocols for Internet Environments," *IEEE Trans. on Communications*, Vol. 43, No. 5, pp. 1947-1950, 1995.
7. W. Juang, "Efficient Password Authenticated Key Agreement Using Smart Cards," *Computers & Security*, Vol. 23, No. 2, pp. 167-173, 2004.
8. W. Juang, "Efficient Multi-server Password Authenticated Key Agreement Using Smart Cards," *IEEE Trans. on Consumer Electronics*, Vol. 50, No. 1, pp. 251-255, 2004.
9. W. Juang, "A Simple and Efficient Conference Scheme for Mobile Communications," the 6th International Workshop on Information Security Applications (WISA2005), Lecture Notes in Computer Science, 3786, pp. 81-95, Springer, New York, 2006.
10. W. Ku and S. Chen, "Weaknesses and Improvements of an Efficient Password Based Remote User Authentication Scheme Using Smart Cards," *IEEE Trans on Consumer Electronics*, Vol. 50, No. 1, pp. 204-207, 2004.
11. M. Kumar, "New Remote User Authentication Scheme Using Smart Cards," *IEEE Trans. Consumer Electron.*, Vol. 50. No. 2, pp. 597-600, 2004.

12. T. Kwon, Y. Park and H. Lee, "Security Analysis and Improvement of the Efficient Password-based Authentication Protocol," *IEEE Commun. Letters*, Vol. 9, No 1, pp. 93-95, 2005.
13. L. Lamport, "Password Authentication with Insecure Communication," *Communications of the ACM*, Vol. 24, pp. 770-772, 1981.
14. A. Lenstra, E. Tromer, A. Shamir, W. Kortsmit, B. Dodson, J. Hughes and P. Leyland, "Factoring Estimates for a 1024-bit RSA Modulus," In Laih, C. (ed.), Advances in Cryptology-AsiaCrypt'03, Lecture Notes in Computer Science, 2894, pp. 55-74, Springer, New York, 2003.
15. R. Merkle, "One Way Hash Functions and DES," In Brassard, G. (ed.), Advances in Cryptology-Crypt'89, Lecture Notes in Computer Science, 435, pp. 428-446, Springer, New York, 1989.
16. NIST FIPS PUB 197, "Announcing the ADVANCED ENCRYPTION STANDARD(AES)," National Institute of Standards and Technology, U. S. Department of Commerce, Nov., 2001.
17. NIST FIPS PUB 180-2, "Secure Hash Standard," National Institute of Standards and Technology, U. S. Department of Commerce, DRAFT, 2004.
18. D. Seo and P. Sweeney, "Simple Authenticated Key Agreement Algorithm," *Electronics Letters*, Vol. 35, pp. 1073 - 1074, 1999.
19. H. Sun, "An Efficient User Authentication Scheme Using Smart Cards," *IEEE Trans. Consumer Electron.*, Vol. 46, No.4, pp. 958-961, 2000.
20. P. Syverson, "A Taxonomy of Replay Attacks," Proc. of Computer Security Foundations Workshop VII, pp. 187-191, 1994.
21. M. Weiser, "Some Computer Science Problems in Ubiquitous Computing," *Communications of the ACM*, Vol. 36, No. 7, pp. 75-84, July 1993.
22. H. Wen, C. Lin and T. Hwang, "Provably Secure Authenticated Key Exchange Protocols for Low Power Computing Clients," *Computers & Security*, in press, 2006.
23. C. Yang and R. Wang, "Cryptanalysis of A User Friendly Remote Authentication Scheme with Smart Cards," *Computer & Security*, Vol. 23, pp. 425-427, 2004.
24. C. Yang, T. Chang and M. Hwang, "Cryptanalysis of Simple Authenticated Key Agreement Protocols," *IEICE Trans. Fundamentals*, Vol. E87-A, No. 8, pp. 2174-2176, 2004.

Single Sign-On and Key Establishment for Ubiquitous Smart Environments

Yuen-Yan Chan[1], Sebastian Fleissner[1], Joseph K. Liu[2], and Jin Li[1,3]

[1] Department of Information Engineering,
Chinese University of Hong Kong,
Shatin, N.T., Hong Kong
{yychan, sfleiss4, jinli}@ie.cuhk.edu.hk

[2] Department of Computer Science,
University of Bristol,
Bristol, UK
liu@cs.bris.ac.uk

[3] School of Mathematics and Computational Science,
Sun Yat-Sen University,
Guangzhou, 510275, P.R. China
sysjinli@yahoo.com.cn

Abstract. In a smart environment, users often need to access multiple service providers. Multiple authentications and key establishments are required as these resources may reside in different security domains. Therefore we are in quest of a solution that combines multiple logins and key exchanges into one single process. Motivated by this need, we propose a scheme for single sign-on and key establishment (SSOKE) for ubiquitous smart environments. We examine the computational model and design considerations for smart environments, and address them in our scheme construction. Security and privacy considerations of our proposal are also provided.

1 Introduction

We[1] are experiencing the era of ubiquitous computing [21], in which computing resources are available everywhere. In ubiquitous computing, there are a lot of computers sharing each of the users. In other words, each user is accessible to many service providers simultaneously. Multiple authentications may be required as these resources may reside in different security domains. Besides, how to establish multiple session keys, each among which is unique between the user and the particular service provider, is also a challenge. Motivated by the problem above, we propose the *Single Sign-On and Key Establishment (SSOKE)* for ubiquitous smart environments. SSOKE combines the two security services: *single sign-on* and *key establishment* into one process. Single

[1] Acknowledgement to Hong Kong Research Grant Council's Earmarked Grants 4232-03E and 4328-02E for sponsorship. Part of the work of the fourth co-author was done while visiting The Chinese University of Hong Kong.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 406–415, 2006.

sign-on (SSO) enables a user to be authenticated once and gain access to resources from multiple security domains; key establishment enables session keys to be agreed and shared between two or more entities. Our scheme is analogous to authenticated key exchange that achieves both functionalities of entity authentication and key establishment. Moreover, our scheme provides client anonymity.

Some work has been done on authentication and key establishment for ubiquitous computing. For example, Verisign proposed Open Authentication as an initiative to provide ubiquitous authentication. The work is echoed by industrial members who together formed the Initiative for Open Authentication (OATH) [15]. Identity federation, which is the establishment of agreements and technologies that make identities and assertions portable across autonomous domains, has also been proposed as a solution for seamless identity management and access control in ubiquitous computing [6, 15]. Other related works include the followings. Stajano *et. al.* proposed the resurrecting duckling approach for establishing transient associations between two devices [17]. Volkmer *et. al.* proposed a low hardware-complexity solution with a Tree Parity Machine Rekeying Architecture [20]. Jenkin *et. al.* [10] proposed the used of one-time pads for secure communication for low-power devices. Issues related to authentication and trust models in ubiquitous computing have also been reported [2, 6, 18, 19].

Previous works of SSO are briefed below. Kerberos Authentication Protocol [4, 8] is an early solution for SSO with key establishment. Users sign into the Kerberos authentication server and ticket granting server which issue tickets to be used in subsequent accesses at other application servers. A session key is conveyed in the application server-specific tickets. Yet warnings on data-integrity of encrypted tickets and authenticators were mentioned in [12]. Recent advances in federated identity management creates a second wave for SSO. To this end, standards and specifications are published by renown organizations, including Security Assertion Markup Language (SAML) published by OASIS [13], as well as the Identity Federation Framework (ID-FF) of the Liberty Alliance Project [16] and the Shibboleth Project of Internet2 [9]. Jeong *et. al.* also proposed an SAML-based architecture for SSO in ubiquitous service environments [3].

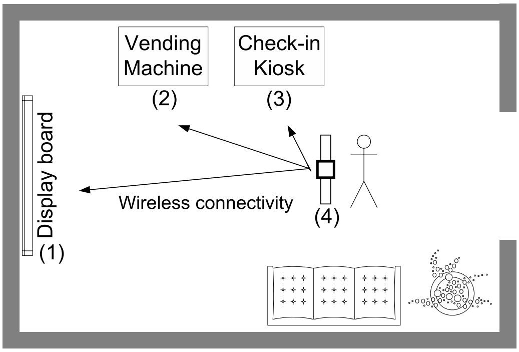
In this paper, we examine the computational model and design considerations for smart environments, and construct a scheme for single sign-on and key establishment based on SAML accordingly. Our scheme is the-first-of-its-kind for ubiquitous computing and it provides client anonymity. Security and privacy considerations of our proposal are also provided. The rest of the paper is organized as follows. We describe the smart environments, provide a conceptual illustration, and discuss the design considerations in Section 2. We propose SSOKE and include its architecture diagram in Section 3. Security and privacy considerations are discussed in Section 4. The paper is concluded in Section 5.

2 Background

2.1 Ubiquitous Computing and Smart Environments

Ubiquitous computing aims at constructing a global computing environment to offer invisible and seamless accesses to computing resources. It leverages on the advances in mobile computing and pervasive computing to provide such environment [5]. The former boosts computing devices and introduces mobility over the wireless infrastructure while the later acquires context from the environment and establishes context-driven computing models dynamically. A *smart environment*, or a *smart space*, is a physical place with embedded computing and network services. In a smart environment, sensors, embedded computing devices, wearable client devices, as well as the wired and wireless network connectivity operate together to provide seamless services to the user.

We construct a conceptual ubiquitous computing scenario in which a traveler arrives at an airport departure lounge and accesses various available services embedded in the smart environment. The scenario is illustrated in Fig. 1.



1. Display board shows flight and destination information
2. Vending machine sells convenience goods and charges at traveler's account
3. Check-in kiosk performs auto check-in for traveler
4. Wearable computing device accesses available services and resources

Fig. 1. Ubiquitous Computing in an Airport Departure Lounge

Computational Model for Smart Environments. With reference to a typical smart environment in a in Fig. 1, we have the following descriptions on its computational model:

1. *Server-Side Infrastructure Model.* The servers exist as computationally enhanced part of a building or a vehicular environment. They are relatively stable and possess normal computational power. In addition, the servers are equipped with sensors and front-end wireless connectivity, with Internet connection supported at the back-end.
2. *Client-Side Infrastructure Model.* In contrast, client-side devices are mobile and limited in computational power. Personal Digital Assistants (PDA) and enhanced mobile handsets are typical examples. The clients can even exist as wearable computing devices which can be carried by users on their person

like watches or belts. Since the wearable devices cannot be bulky, their computational power is further limited. Client devices are also equipped with wireless connectivity, and probably, sensors that help seeking services available around.
3. *Volatile Client-Server Relationship.* The client-server relationship in ubiquitous computing is *volatile* and intangible. Take the airport departure lounge as an example: thousands of travelers may arrive and depart at the airport everyday. Relationship between the users and the service providers change dynamically and unpredictably. Such relationship may even be *spontaneous* as travelers may origin from different states and countries such that it is infeasible to maintain a database of their records.

Design Considerations. Based on the computational model described above, we summarize the design considerations for security services in ubiquitous smart environments below:

1. *Limited computational power at client side.* Because of the size and portability of the mobile devices, client-side computational power is limited in terms of processor speed and storage capacity. Therefore, heavy cryptographic computations such as private key decryption should be avoided.
2. *Constrained bandwidth between clients and servers.* Client and server devices communicate via short-range wireless connection which has limited bandwidth. Therefore the number of messages exchanged between the client and server as well as the corresponding message length should be restricted.
3. *Stable servers with normal back-end connectivity.* Despite the limitations of the client devices, server devices in smart environments are relatively stable and can have computational power and storage capacity as conventional computers. Also, they are connected to the Internet through wired network at back-ends. Therefore they can afford heavy cryptographic computations as well as relatively lengthy messages with other servers.
4. *Absence of long-term client profiles at smart service providers.* Because of the volatility in client-server relationship, no long-term user profiles or keying materials are kept at the smart service providers. Therefore security services should not rely on symmetric keys shared between clients and servers.

In short, computation in smart environments is *asymmetric* that the client side is restricted but the server side is not. Therefore hybrid approach, which requires different computational loads on server and client, is suggested.

2.2 Identity Federation and Single Sign-On

Ubiquitous computing services often involve multiple providers. Various user profiles are involved in the course of the services provision, each requiring independent authentication and credential submission. This violates the seamless end-user experience in ubiquitous computing. *Identity federation* and *single sign-on* are two emerging security services in electronic business for idenity and access management. With suitable adaptations, they can be applied to ubiqtuitous computing and smart environments.

Identity Federation. *Federated identity* refers to the agreements, standards, and technologies that make identity and entitlements portable across autonomous domains [7]. It is analogous to a passport, where one country provides citizens with a credential that is trusted and accepted as proof of identity by other countries [11]. Identity federation is the establishment of the federated identities and corresponding agreements, cryptographic trusts, and user identifiers among organizations. The most influential industrial driver to identity federation is Project Liberty [16], a consortium composed of over 170 organizations from around the globe. Identity federation is also a cornerstone of single sign-on.

Single Sign-On. Single sign-on is a technique that enables a user to authenticate once and gain access to the resources of multiple systems, which may resides in different security domains. Typically single sign-on involves three principals:

- *User.* Who accesses the network and makes use of the system resources for any purposes. In practice, the user is often represented by a *user agent.*
- *Service provider.* Who offers restricted services which are only available to authenticated principals.
- *Identity provider.* Who creates, maintains, and manages identity information for principals and provides principal authentication to other service providers.

Identity federation among the user, service provider(s), and identity provider is required before SSO takes place. When a user whose identity is federated to the identity provider requests service from the service provider, who also federated to the identity provider, the service provider sends an authentication request to the identity provider. Based on the authentication status of the user, the identity provider produces security assertions of the user and send back to the service provider. Then the service provider accepts or rejects the user's request based on the security assertions as well as its own authorization policies. The SSO process can involve no user authentication so that it is transparent to the user. This feature is desirable to smart services in ubiquitous computing.

SAML. Security Assertion Markup Language (SAML) is an industrial facilitator of identity federation and single sign-on. It is established by the Organization for Advancement of Structured Information Standards (OASIS). Its first version, SAML V1.0 [13], was published in November, 2002. Since then, OASIS has released subsequent versions of SAML and the latest one is SAML V2.0 [14] that released in March, 2005. Today, SAML has been broadly implemented in major enterprise Web server and application server products. The SAML Standard consists of a set of XML schemas and specifications, which together define how to construct, interchange, interpret, and extend security assertions for a variety of purposes. The major ones include web Single Sign-On (web SSO), identity federation, and attribute-based authorization. There are specifications that being built on-top of the SAML core framework and make use of the adaptations of the SAML bindings and profiles. These include the Identity Federation Framework (ID-FF) of the Liberty Alliance Project [16] and the Shibboleth Project of Internet2 [9]. Both of these specifications define the SSO functionality.

3 Single Sign-On and Key Establishment

3.1 Preliminaries

Notations. We introduce the notations to be used in the schemes:

1. Principals: U denotes a user. SP denotes a service provider, IdP denotes an identity provider. U and IdP maintain a synchronized counter CTR.
2. Expressions: `<.>` indicates an XML element. $E_X(m)$ denotes a message (or message segment) m being encrypted with the encryption key of an entity X. $Sig_X(m)$ denotes a message (or message segment) m being signed with the signing key of an entity X. $\mathbf{A} = A(1 \ldots n) \in \mathbb{Z}_n$ denotes an ordered array of n variable A. $A(i)$ denotes the i^{th} element in the array $\mathbf{A}$. $f : \mathbb{Z}_n \mapsto \mathbb{Z}_q$ is a collision free one-way function where q is a security parameter.
3. Parameters: sk denotes the secret key shared between U and IdP, U, $SEED$ $\in \mathbb{Z}_n$ denote the user's identity alias and key seed respectively, $\mathbf{U} = U(1 \ldots n)$ and $\mathbf{SEED} = SEED(1 \ldots n)$ denote an ordered array of n alias and key seeds respectively. $SessionKey$ is the resulting session key in the single sign-on and key establishment procedure.
4. SAML elements: `<AuthnRequest>`, `<Request>`, and `<Assertion>` denote the authentication request, response, and assertion elements respectively. `ID`, `IssueInstant`, and `Status` denotes the message ID, timestamp, and authentication status respectively.
5. Custom abbreviations: `req`, `res` and `asrt` abbreviates `AuthnRequest`, `Response`, and `Assertion` respectively.

Architecture. The architecture for SSOKE in a smart environment is depicted in Fig. 2. It consists of a user (U), an identity provider (IdP), and multiple service providers (SP). The SPs may belong to different network domains. U connects to SPs via wireless network, and a secure channel between SPs and IdP is established over the wired Internet.

3.2 SSOKE

SSOKE comprises two procedures. In *Federation and Key Seed Vector Distribution*, IdP federates (registers) U. IdP also generates and distributes the key seed vector $\mathbf{SEED} = SEED(1 \ldots n)$ and identity alias vector $\mathbf{U} = U(1 \ldots n)$ to U through some secure channels. A counter between U and IdP, CTR, is also initialized. In *Single Sign-On and Key Establishment*, U who is federated to IdP requests services from SP. SP then sends an authentication request to IdP, who produces an authentication assertion of U for SP. The assertion conveys a session key seed $SEED(CTR)$ that contributes to the session key $SessionKey$ to be shared between SP and U.

Federation and Key Seed Vector Distribution. SUMMARY: IdP federates U and performs set up for subsequent single sign-on and key establishment procedures (Fig. 3).

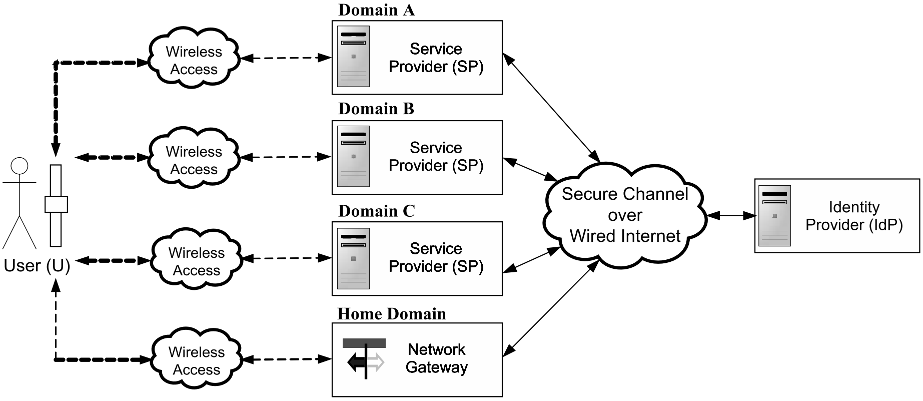


Fig. 2. Single Sign-On and Key Establishment Architecture

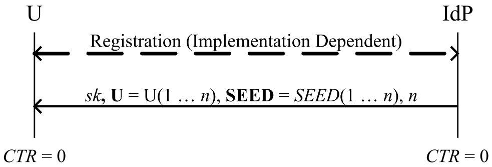

U
IdP
Registration (Implementation Dependent)
sk, U = U(1 … n), SEED = SEED(1 … n), n
CTR = 0
CTR = 0

Fig. 3. Federation and Key Seed Vector Distribution

1. *One-time setup.*
   (a) U registers to IdP with implementation dependent mechanisms.
   (b) A secure channel is established between U and IdP.
2. *Protocol message.*
   U ← IdP: $sk, \mathbf{U}, \mathbf{SEED}, n$ (1.1)
3. *Protocol actions.*
   (a) *Step 1.* IdP generates the identity alias vector $\mathbf{U} = U(1 \ldots n)$ and key seed vector $\mathbf{SEED} = SEED(1 \ldots n)$.
   (b) *Step 2.* IdP sends $sk, \mathbf{U}, \mathbf{SEED}$, and n to U.
   (c) *Step 3.* U and IdP initialize CTR to 0 respectively.

Single Sign-On and Key Establishment. SUMMARY: SP establishes authentication context with U through the assertion from IdP. At the same time, a common session key $SessionKey$ is agreed between U and SP (Fig. 4).

1. *One-time setup.*
   (a) Secure channel established between SP and IdP.
   (b) U has its identity federated between IdP and SP by business agreement.
   (c) U and IdP performed the Federation and Key Seed Vector Distribution procedure.

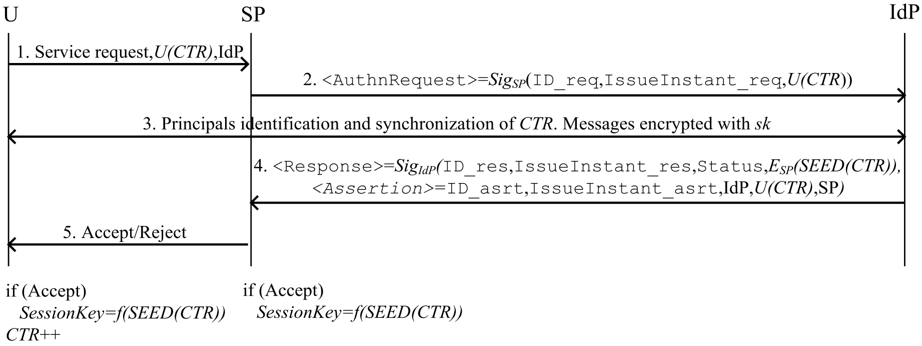

U
SP
IdP
1. Service request,U(CTR),IdP
2. <AuthnRequest>=Sig_SP(ID_req,IssueInstant_req,U(CTR))
3. Principals identification and synchronization of CTR. Messages encrypted with sk
4. <Response>=Sig_IdP(ID_res,IssueInstant_res,Status,E_SP(SEED(CTR)),
<Assertion>=ID_asrt,IssueInstant_asrt,IdP,U(CTR),SP)
5. Accept/Reject
if (Accept)
SessionKey=f(SEED(CTR))
CTR++
if (Accept)
SessionKey=f(SEED(CTR))

Fig. 4. Single Sign-On and Key Establishment

2. *Protocol messages.*
   U → SP: Service request, $U(CTR)$, IdP (2.1)
   SP → IdP: `<AuthnRequest>` (2.2)
   U ↔ IdP: Principals identification and synchronization of CTR (2.3)
   SP ← IdP: `<Response>` (2.4)
   U ← SP: Accept / Reject (2.5)
3. *Protocol actions.*
   (a) *Step 1.* U makes a service request along with its identity alias and the identity of IdP at SP without a security context.
   (b) *Step 2.* SP issues an authentication request `<AuthnRequest>` to IdP.
   (c) *Step 3.* IdP identifies U by looking up $U(CTR)^2$. Upon successful principal identification, U and IdP synchronize the value of CTR. Messages exchanged in this step are encrypted with sk.
   (d) *Step 4.* According to the authentication result of Step 3, IdP encrypts $SEED(CTR)$ using SP's encryption key. IdP forms, signs and sends `<Response>` that conveys `<Assertion>` to SP.
   (e) *Step 5.* Based on the status and the signed assertion, SP either accepts U's service request or rejects it. In either case, U increases the value of CTR by 1. If the request is accepted, both SP and U generates $SessionKey = f(SEED(CTR))$ respectively.

4 Discussions

We highlight the discussions in the followings:

1. *Authentication.* During SSOKE, registered client is authenticated by IdP at Step 3 via implementation dependent mechanisms, while SP and IdP authenticate each others by means of the digital signatures. SP authenticates U by the assertion from IdP.

[2] When necessary, implementation dependent mechanisms (such as challenge and response of sk) can be used to further authenticate U.

2. *Client anonymity.* Client anonymity at SP is supported as U does not present its identity during SSOKE. Instead, a one-time identity alias $U(CTR)$ is used. However, for critical services such as account withdrawals, a real user identity can be used instead.
3. *Key establishment.* Upon the completion of SSOKE, both U and SP share $SessionKey$. Unfortunately, similar to authentication and key agreement in 3GPP UMTS [1], IdP (*c.f.* Home Location Register in UMTS) also has access to this key. This leaves an open problem in our paper.
4. *Efficiency.* Our scheme requires no heavy computations at client side. Rather, it shifts such computations as signature verifications and private key decryptions to SP and IdP.
5. *Attacks and their preventions.* We briefly highlight the prevention of common attacks in SSO and key establishment protocols. Replay attack is prevented by the use of message identifiers `ID`s and timestamps `IssueInstant`. Impersonation of U by replaying message 2.1 is detectable during CTR synchronization in step 3 of SSOKE. Eavesdropping can be prevented by symmetrically encrypt the messages sent by U with sk, together with the secure channel established between SP and IdP. Man-in-the-middle between SP and IdP is prevented by the secure channel, while man-in-the-middle between U and SP (for example, adversaries may act as dishonest SP) can be detected by implementation dependent mechanisms during Step 3 of SSOKE.

5 Conclusion

In this paper, we have briefly reviewed on a number of authentication and key establishment protocols for ubiquitous computing, as well as a number of works and standards on single sign-on. we have discussed the computational model and design considerations for smart environments with the illustration of a conceptual situation. We have also constructed a scheme for single sign-on and key establishment based on SAML with accordance to the design considerations, and our scheme provides client anonymity at service providers. We have also considered on its security and privacy and discussed the preventions of various attacks common in single sign-on and key establishment protocols. An open problem of the knowledge of the session key of the identity provider, which is similar to that in 3GPP UMTS authentication and key agreement, is left for future research.

References

1. 3GPP TS 33.102. *3G Security; Security Architecture (v6)*, Sept 2005.
2. J. Bardram. The trouble with login – on usability and computer security in ubiquitous computing. *Personal and Ubiquitous Computing*, July 2005.
3. J. Jeong et. al. A study on the xml-based single sign-on system supporting mobile and ubiquitous service environments. In *International Conference on Embedded and Ubiquitous Computing*, pages 903–913, August 2004.
4. S. Miller et. al. Kerberos authentication and authorization system. Technical report, Project Athena, Massachusetts Institute of Technology, 1987.

5. S. Singh et. al. Ubiquitous computing: connecting pervasive computing through semantic web. In *Information Systems and E-Business Management*. Springer-Verlag, 2005.
6. T. Walter et. al. Security and trust issues in ubiquitous environments - the business-to-employee dimension. In *SAINT 2004 Workshops*, pages 696 – 701, 2004.
7. Burton Group. Burton group federated identity. Web Site, 2005.
8. IETF RFC 1510. *The Kerberos Network Authentication Service (v5)*, Sept 1993.
9. Internet2. http://www.internet2.edu/.
10. M. Jenkin and P. Dymond. One-time pads for secure communication in ubiquitous computing. In *Proceedings of IASTED*, 2004.
11. RSA Security Ireland Limited. Secure business-to-business single sign-on (b2b sso) based on federated identity management. Technical report, RSA Security Inc., 2004.
12. W. Mao. *Mondern Cryptography: Theory and Practice*. Prentice-Hall PTR, Upper Saddle River, NJ, May 2004.
13. OASIS SSTC. *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML)*, November 2002.
14. OASIS SSTC. *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0*, 2005.
15. OATH. *OATH Reference Architecture Version 1.0*, 2005.
16. Liberty Alliance Project. http://www.projectliberty.org/.
17. F. Stajano and R. Anderson. The resurrecting duckling: Security issues for ad-hoc wireless networks. In *Security Protocols Workshop*, pages 172–194, 1999.
18. F. Stajano and R. Anderson. The resurrecting duckling: security issues for ubiquitous computing. *Computer*, 35(4):22–26, April 2002.
19. Upkar Varshney. Network access and security issues in ubiquitous computing. In *Workshop on Ubiquitous Computing Environment*, October 2003.
20. M. Volkmer and S. Wallner. A key establishment ip-core for ubiquitous computing. In *DEXA Workshops*, pages 241–245, 2005.
21. M. Weiser and J. S. Brown. The coming age of calm technology. In *Beyond Calculation: The Next Firty Years of Computing*, pages 75–85. Copernicus, New York, NY, 1997.

A Light Weight Authentication Protocol for Digital Home Networks

Ilsun You[1] and Eun-Sun Jung[2]

[1] Department of Information Science, Korean Bible University,
205 Sanggye-7 Dong, Nowon-ku, Seoul, 139-791, South Korea
isyou@bible.ac.kr
[2] Communication LAB, Samsung Advanced Institute of Technology,
San 14-1, Nongseo-Dong, Kihueng-Gu, Yongin-Si, Kyunggi-Do 449-712, Korea
eun-sun.jung@samsung.com

Abstract. We study user authentication protocols that allow user to remotely access and control home appliances through home gateway. In particular, we explore the S/Key user authentication scheme, a widely known one-time password system. Earlier studies show that S/Key is vulnerable to server spoofing, replay, and off-line dictionary attacks. Several researchers have proposed various solutions to prevent such attacks. However, we show that these enhancements are still vulnerable to another security attacks and propose a scheme that defends such attacks.

1 Introduction

With the proliferation of the Internet technology and electronic devices, digital home network has received significant attention in the last few decades [1]. While the home network technologies provide new ways to access and manage home equipments, they also create numerous challenges. One of the main challenges in home networks is security issue due to the rapid deployment of wireless networks.

As shown in Fig. 1, a typical home network is consisted of home gateway, home appliances, mobile devices, and service providers. Among these components, home gateway plays an important role in connecting the home network to external network. Also, it provides users with several services such as routing, firewall, and access control. Remote access control is an important service in the design of home networks. It allows residential users to remotely access and control home appliances such as TVs, lights, washing machines, and refrigerators. For example, they can turn on or off their home appliances from their offices. However, despite such conveniences, the remote control service causes digital home networks to have various security threats such as masquerade, denial of service attacks, etc. Furthermore, handheld devices are often connected to digital home networks by wireless links and the links are especially vulnerable to passive eavesdropping, active replay attacks, and other active attacks. Therefore, it is necessary to provide strong security services in digital home networks. In particular, user authentication is a key service required for remote access control.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 416 – 423, 2006.

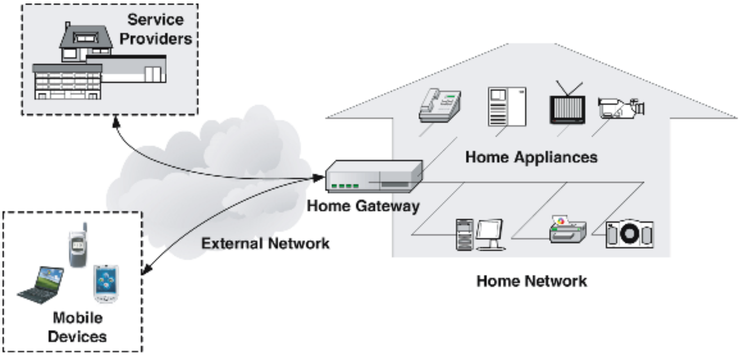


Fig. 1. Digital Home Network Structure

When we design a user authentication scheme for home networks, we should consider the following requirements:

- User authentication should be strong enough to protect the home gateway from eavesdropping and various active security attacks.
- Mutual authentication should be provided - the mechanism should be able to verify the server as well as the user (or client).
- Authentication scheme should be light weight. Since user devices may be mobile and resource constrained, we consider lightweight cryptographic operations such as hash function to provide user authentication rather than relying on expensive *asymmetric* cryptographic operations. Especially, on CPU-limited devices, asymmetric cryptographic operations are much slower than symmetric cryptographic operations. Moreover, it is computationally expensive or energy-intensive to perform asymmetric cryptographic operations.

Based on these observations, we investigate a well known one-time password system, S/Key [2,3]. As earlier studies show, S/Key is vulnerable to server spoofing, replay, and off-line dictionary attacks [4]. Several enhancements, including a work by Lee-Chen, have been proposed [5-8] but they are still vulnerable to other security attacks. In this paper, we propose an enhanced one-time password authentication protocol for digital home networks.

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 discusses the Lee-Chen's scheme, an improved S/Key protocol, and describes its weaknesses. Section 4 presents our proposed authentication scheme. Section 5 analyzes the proposed scheme and section 6 concludes the paper.

2 Related Work

The S/KEY one-time password scheme is designed to protect a system against replay or eavesdropping attacks [2,3]. With S/Key, the user's secret pass-phrase never needs to cross the network at any time such as during the authentication or pass-phrase changes. Moreover, no secret information needs to be stored on any system, including the server being protected. Although the S/KEY scheme protects a system against passive attacks based on replaying captured reusable passwords, it is vulnerable to server spoofing, preplay, and off-line dictionary attacks [4,6].

Several researchers have been conducted to solve these drawbacks of the S/KEY scheme [4-8]. Mitchell and Chen propose two possible solutions to resist against server spoofing and preplay attacks. One is to locally store the predictable challenge in a client so that a server has no need to send the challenge in every login. The other is to digitally sign the predictable challenge.

Yen and Liao propose a scheme that uses a shared tamper resistant cryptographic token, which includes a SEED, to prevent off-line dictionary attacks.

Recently, Yeh, Shen and Hwang propose a one-time password authentication scheme which enhances the S/KEY scheme to resist against the above attacks. The scheme uses smart cards to securely preserve a pre-shared secret, SEED, and simplify the user login process. Also, it provides a session key to enable confidential communication over the network. However, since the scheme uses user's weak pass-phrase and utilizes SEED as a pre-shared secret, the exposure of the SEED causes the scheme to retain the flaws of the S/KEY scheme [9]. Consequently, the scheme cannot achieve the strength of the S/KEY scheme that no secret information needs to be stored on the server. In addition, it cannot defend against server compromises and is vulnerable to several attacks such as stolen-verifier attacks, denial of service attacks and Denning-Sacco attacks [9-11]. Obviously, this scheme has drawbacks and cannot satisfy high-level security, which remote control service requires.

In [8], the S/KEY based authentication scheme using a server-side public key is proposed. However, the scheme is vulnerable to denial of service attacks. Also, it does not provide a way to verify the server's public key.

Lee and Chen propose an improvement on Yeh-Shen-Whang's authentication scheme to prevent its vulnerability from stolen verifier attacks [5,6]. With such improved security, their scheme still provides the same efficiency as Yeh-Shen-Whang's protocol, so it is desirable for low-power mobile devices. We review Lee-Chen's scheme in more detail in the following section.

3 Review of Lee-Chen's Scheme

Lee-Chen's scheme is composed of three stages: registration, login and authentication stages. This section reviews the scheme and analyzes its drawbacks. It is assumed that the mobile device initially receives *SEED* from the server by out-of-band distribution.

3.1 Notation

- U denotes the user
- S denotes the server
- ID denotes the user's identifier
- x denotes the secret key of the server
- K denotes the user's secret key
- $H()$ denotes one-way hash function
- $SEED$ denotes a pre-shared secret of the mobile device and the server
  $SEED = H(ID \oplus x)$
- || denotes a concatenate operation

3.2 Registration Stage

Fig. 2 shows the registration stage, where the user's login information is synchronized between a user and a server.

(1) $S \rightarrow U : N, H(SEED \oplus N) \oplus SK, H(SK)$
(2) $U \rightarrow S : P_0 \oplus SK$

- It is assumed that the mobile device contains *SEED* beforehand
- D is a random number generated by the server
- N denotes a permitted number of login times
- TS is a timestamp
- $SK = D || TS$
- $P_0 = H^N(K \oplus SEED)$

Fig. 2. Registration Stage of Lee-Chen's Scheme

To register the user, the server randomly generates D, computes $H(SK)$ and $H(SEED \oplus N)$, and performs an XOR operation on $H(SEED \oplus N)$ and SK. Then, it sends the user $H(SEED \oplus N) \oplus SK$ and $H(SK)$ along with N, a permitted number of login times. Upon receiving them, the user computes $H(SEED \oplus N)$ and applies an XOR operation to the result and the received $H(SEED \oplus N) \oplus SK$ to extract SK. If the hash value of the extracted SK is equal to $H(SK)$, the user computes and sends $P_0 \oplus SK$. As a result, the login information P_0 and N are stored at the server. In addition, the smart card stores P_0.

3.3 Login and Authentication Stages

Fig. 3 shows the login stage, which provides mutual authentication between a user and a server.

(1) $S \rightarrow U : Ci, H(SEED \oplus Ci) \oplus SKi, H(SKi) \oplus P_{i-1}$
(2) $U \rightarrow S : P_i \oplus SKi$

- $Ci = N-i$
- Di is a large random number generated by the server
- TSi is a timestamp
- $SKi = Di || TSi$
- $P_i = H^{Ci}(K \oplus SEED)$

Fig. 3. Login Stage of Lee-Chen's Scheme

When receiving the message of step (1), the user first computes *H(SEED ⊕ Ci)* and extracts *SKi* by performing an XOR operation on *H(SEED ⊕ Ci) ⊕ SKi* and the computed value. Next, the user hashes *SKi* and compares the hashed value with *H(SKi)* to authenticate the server. If the server is valid, it computes and sends $P_i \oplus SKi$. On the receipt of the message of step (2), the server extracts P_i and verifies if the hash value of P_i is equal to the stored P_{i-1}. If the verification is positive, the server can ensure that the user is valid. Finally, the server updates the last one-time password P_{i-1} with P_i and the count value with *Ci*.

3.4 Analysis

Although Lee-Chen's scheme strengthens the security of Yeh-Shen-Whang's scheme and keeps the efficiency of the scheme, it results in the following drawbacks:

Denial of Service Attacks
A denial of service (DoS) attack is an offensive action where an attacker attempts to prevent legitimate users from accessing information or services [11]. In Lee-Chen's scheme, an attacker can modify messages during the registration stage without being detected, which results in desynchronization between a server and a client. A DoS attack can be launched as follows. Assume that an attacker can eavesdrop, record, inject, re-order, and re-send (altered) messages. During the registration stage, the attacker can replace $P_0 \oplus D$ of step (2) with an equal-sized random number, *R*. In this case, upon receiving the modified message, the server computes the initial password P_0' by performing an XOR operation on the values of *R* and *SK*. Since P_0' is invalid, the server and the client become desynchronized. This causes the server to deny the client access during the login and authentication stages. Therefore, any attacker can easily mount DoS attacks without using any cryptographic methods.

Compromise of Past Session Keys by Stolen Password
After the authentication stage, *SKi*, generated by the server, can be used as a session key. In this case, a compromise of the *i*th password, P_i, allows an attacker to compromise past session keys. Assume that an attacker records all messages exchanged between the server and the user during a session and manages to obtain the *i*th password, P_i. Then, the attacker can mount the following attack by using P_i:

```
- [input] $P_i$: the ith one time password
- [input] R: an array of the step (2) messages recorded during the login stage
              i.e.) $R[t] = P_t \oplus SKt$
- [output] SK: an array of session keys

(1) $otp = P_i$
(2) for( t = i; t > 0; t--)
(3) {
(4)     $SK[t] = R[t] \oplus otp$
(5)     $otp = H(otp)$
(6) }
```

4 Proposed Protocol

We now present our enhanced one-time password authentication scheme, which improves the drawbacks of Lee-Chen's scheme discussed in the previous section.

4.1 Registration Stage

Fig. 4 illustrates the registration stage of our proposed scheme. Unlike the Lee-Chen's scheme, the user sends $H(P_0 \| SK)$ as well as $P_0 \oplus SK$ to the server at step (2). When receiving the two values, the server first extracts P_0 and computes $H(P_0 \| SK)$. Then, it compares the computed hash value with the received one. If they are equal, the server can ensure that P_0 is valid. Thus, $H(P_0 \| SK)$ enables the server to defend against DoS attacks.

(1) $S \rightarrow U : N, H(SEED \oplus N) \oplus SK, H(SK)$
(2) $U \rightarrow S : P_0 \oplus SK, H(P_0 \| SK)$

Fig. 4. Registration Stage of the Proposed Protocol

4.2 Login and Authentication Stages

Fig. 5 shows the login stage, where the message of step (2) includes $P_i \oplus H(SKi)$ instead of $P_i \oplus SKi$. Including $H(SKi)$ inside the message can prevent the attacker, who knows P_i, from compromising past session keys. When receiving the message, the server computes $H(SKi)$ and extracts P_i by applying an XOR operation to the received $P_i \oplus H(SKi)$ and the computed value. Next, it verifies P_i to authenticate the user. As in Lee-Chen's scheme, if P_i is valid, the server updates the last one-time password P_{i-1} with P_i and the count value with *Ci*.

(1) $S \rightarrow U : Ci, H(SEED \oplus Ci) \oplus SKi, H(SKi) \oplus P_{i-1}$
(2) $U \rightarrow S : P_i \oplus H(SKi)$

Fig. 5. Login Stage of the Proposed Protocol

4.3 Analysis

We now analyze the proposed protocol in terms of security and performance.

4.3.1 Security

Denial of Service Attacks

To defend against DoS attacks, the proposed protocol adds $H(P_0 \| SK)$ to the step (2) message of the registration stage. When receiving the step (2) message, the server extracts P_0 by applying an XOR operation to $P_0 \oplus SK$ and SK and verifies if the value of $H(P_0 \| SK)$ is valid. If the verification is positive, it can ensure that P_0 is not altered. Thus, the added value $H(P_0 \| SK)$ prevents the server and the client from being desynchronized.

Compromise of Past Session Keys by Stolen Password

The proposed protocol replaces $P_i \oplus SKi$ with $P_i \oplus H(SKi)$ in the step (2) of the login stage. Due to the XOR operation on P_i and $H(SKi)$, an attacker cannot obtain SKi even if it has a knowledge of P_i. Therefore, a compromise of the ith password P_i does not guarantee a compromise of the past session keys.

Table 1. Performance Comparison of Proposed Protocol and Lee-Chen's Protocol

Step	Proposed Protocol	Lee-Chen's Protocol
Registration 1	4× (Hash + XOR)	4× (Hash + XOR)
Registration 2	2× (Hash + XOR)	2× XOR
Login 1	4×Hash + 6×XOR	4×Hash + 6×XOR
Login 2	(N-i+1) ×Hash + 1×XOR	(N-i) ×Hash +1×XOR
Authentication	2×Hash + 1×XOR	1×Hash + 1×XOR
Total	(N-i+13) ×Hash + 14×XOR	(N-i+9) ×Hash + 14×XOR

* Hash denotes a hash operation

* XOR denotes an exclusive-OR operation

4.3.2 Performance

Table 1 compares the performance of the proposed protocol with that of Lee-Chen's protocol. As shown in the table, the proposed protocol has only 4 additional hash operations to solve the problems mentioned in the section 3.4.

5 Conclusion

Despite a fast-growing interest in the design of home network, security threat has been a main obstacle to make home network a part of our daily lives. In this paper, we propose an enhanced one-time password authentication protocol for user devices in digital home networks. Since user devices tend to be mobile and resource constrained, we consider the S/Key scheme and its variants, which uses lightweight cryptographic operations such as exclusive-OR and Hash function. Lee-Chen's protocol based on S/Key provides an efficient and secure authentication that defends against various attacks. It also introduces a session key to enable confidential communication between a server and a client. However, we have found that it is vulnerable to DoS attacks and also allows a compromise of past session keys. Our proposed protocol resolves the deficiencies.

References

1. H. Sun, "Home Networking," Mitsubishi Electric Research Laboratories, 2004, http://www.merl.com/projects/hmnt/
2. N. Haller, "The S/KEY One-time Password," RFC 1760, Feb. 1995.

3. N. Haller, C. Metz, P. Nesser and M. Straw, "A One-time Password System," RFC 2289, Feb. 1998.
4. C. J. Mitchell and L. Chen, "Comments on the S/KEY User Authentication Scheme," ACM Operating Systems Review, vol.30, no.4, pp.12-16, Oct. 1996.
5. T. C. Yeh, H. Y. Shen and J. J. Hwang, "A Secure One-Time Password Authentication Scheme Using Smart Cards," IEICE Transaction on Communication, vol.E85-B, no.11, pp.2515-2518, Nov. 2002.
6. N. Y. Lee and J. C. Chen, "Improvement of One-Time Password Authentication Scheme Using Smart Cards," IEICE Transaction on Communication, vol. E88-B no.9, pp.3765-3767, Sept. 2005.
7. S. M. Yen and K. H. Liao, "Shared Authentication Token Secure against Replay and Weak Key Attacks," Information Processing Letters, vol.62, pp.77-80, 1997.
8. I. You and K. Cho, "A S/KEY Based Secure Authentication Protocol Using Public Key Cryptography," The KIPS Transactions: Part C, Vol. 10-C, No.6, Feb. 2003.
9. I. You and K. Cho, "Comments on YEH-SHEN-HWANG's One-Time Password Authentication Scheme," IEICE Transaction on Communication, vol. E88-B, no. 2 pp.751-753, Feb. 2005.
10. D. Denning and G. Sacco, "Timestamps in Key Distribution Systems," Communications of the ACM, vol.24, no.8, pp.533-536, Aug. 1981.
11. S. Kim, B. Kim, S. Park and S. Yen, "Comments on Password-Based Private Key Download Protocol of NDSS'99," Electronics Letters, vol.35, no.22, pp.1937-1938, 1999.

Smart Home Microcontroller: Telephone Interfacing

Chee-Seng Leong and Bok-Min Goi*

Faculty of Engineering, Multimedia University,
63100 Cyberjaya, Selangor, Malaysia
bmgoi@mmu.edu.my

Abstract. In this paper, we present the results of the effort to design and produce a smart home system, PhoneTech. PhoneTech has been designed and built utilizing the telephone network to enable users to remotely control an array of automated home electronic devices by entering a series of commands through phone. The aim of PhoneTech is to provide the users with a better home life experience without overpowering them with complex technologies while keeping the home life as normal as possible.

Keywords: Smart home, automation, DTMF, embedded systems and home-network applications.

1 Introduction

A new low cost smart home system that would enable the user to remotely activate or deactivate electronic home appliances from a very long distance has been designed and developed. This system is named as PhoneTech. The user can be anywhere in the world, and the phone can be any touch-tone phone or cell phone. The basic idea of this project is to take advantage of the vast network of telephone lines and the proliferation of cell phones to extend human's reach and possibilities [1]. The system, through telephone networks, connects the user to home appliances at home and gives him/her the ability to switch them ON or OFF. The user dials the home telephone number like an ordinary telephone call. The telephone at home rings and if nobody picks the call up to 3 rings (or any value set by user), then the system picks up the call. The system then asks for the password. The user enters the password and if no password is entered within a certain time, the system will hang up. Once the password is entered, the user will be offered a voice menu and asked to choose from that menu. The user chooses an item from the menu by pressing a button on the phone keypad. Pressing a button on the phone generates a DTMF signal [2] which, through the telephone network, will reach the system at home. The system will recognize the received signal and based on that will switch on/off the chosen appliance either through a physical wiring connection or infrared connection. The system

* The second author acknowledges the Malaysia IRPA grant (04-99-01-00003-EAR).

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 424–431, 2006.

transmits data serially to the PC so that the user will be able to keep track the call details made by the user. Then, the PC triggers the GSM mobile phone to send a SMS to the pre-defined telephone number in order to notify him/her the action taken on the system. The designed system has many components such as microcontroller, DTMF decoder, telephone interfacing circuit, voice chip and etc.

2 Design Goals

2.1 Specification

- **Inputs:**
  There are only 2 inputs to the system. One of them is connected to the phone jack while another one is connected to a switch which is used to imitate the operation of an alarm.
- **Outputs:**
  The system controls 2 appliances in this prototype. It can be developed to control more number of appliances. In this system, one of the control signals is physically wired to the appliance while the other one is connected remotely through infrared connection. The system also offers a voice message for users. These audio signals travel over the same phone line through which the input comes to the system. The system also communicates with the PC serially. Each action taken by the user will be kept track by the system and then recorded down by the PC. Then the system will alert him/her on the action taken by the user through SMS. An auto-dialing feature is implemented to alert the user for emergency purposes.
- **Functions of the system:**
  - Switch on/off two electric appliances.
  - Requires a four-digit password - To provide security to prevent breaking into the system.
  - Allows the user to change password, ring detection count, and password attempt.
  - Record down the action taken by user in PC.
  - Send SMS to notify the action taken by user.
  - Auto-dialing for emergency purposes.
  - Provide voice message to inform user of:
    - Menu options available.
    - Acknowledgment of data received.
- **Power supply:** The system used standard +5V and -5V DC.

2.2 Applications

`PhoneTech` can boast a variety of convenience [3]. Remote on/off control may be given to electric appliances such as slow rice cooker, exterior lighting and garage heater. Video buffs could interface to their VCR remote control inputs and record T.V. shows with a few keystrokes of their telephones. Scheduled changes or unexpected broadcast could be captured from any remote location featuring a touch tone phone (even a public phone will do). Security system could be controlled and a microphone could be switched on for remote audio monitoring. Interfacing to a home computer widens the system's data communication and data processing capability [4].

3 Design and Implementation

We firstly provide an overview of the whole system in the form of block diagram, as shown in Figure 1. The function of each block is summarized below:

- **Protection circuit:** Protect the system from telephone line transients.
- **On/Off-hook circuit:** Pick up or hang up the call.
- **Ring detection circuit:** Detect incoming ringing signal.
- **2/4 wire converter:** Split bidirectional audio from the balanced telephone line into separate single ended transmit and receive paths.
- **DTMF decoder:** Decode DTMF signals and represents them in a sequence of four bits.
- **Voice chip:** Play pre-recorded messages for users.
- **Microcontroller:** Control all logic operations of the system.

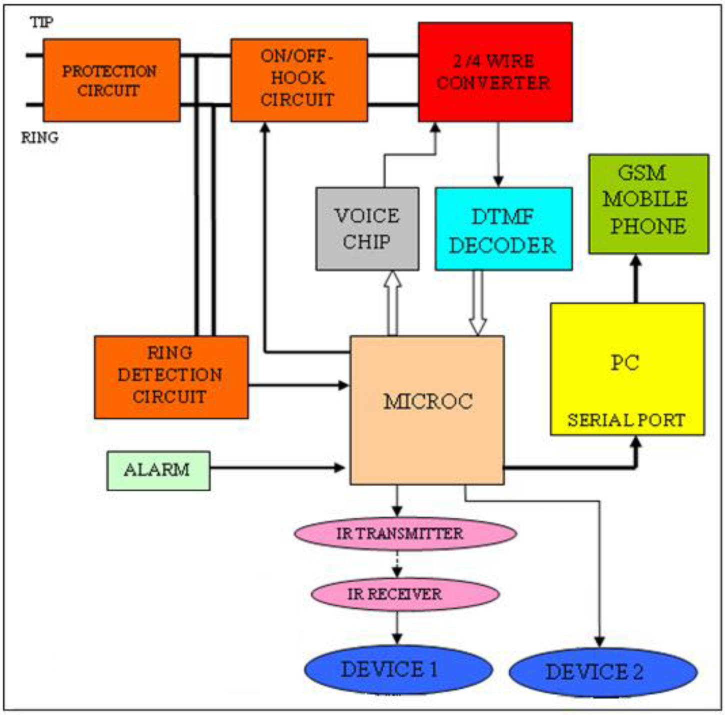


Fig. 1. Block diagram of whole system

- **IR transmitter & receiver:** Provide wireless remote control feature.
- **PC (personal computer):** Communicate serially with the system to display call details on monitor. Originate the sending of SMS and auto-dialing.
- **GSM mobile phone:** GSM modem to send SMS and auto-dialing.
- **Alarm:** A switch used to imitate the operation of alarm.

3.1 Telephone Line Interfacing Circuit

The functions of telephone interfacing circuit are to detect incoming ringing signals, take the line and hang up. It is consists of ring detection circuit and on/off-hook circuit. It is also equipped with protection circuit to protect the system from high voltage transient that might occur on it [5].

Protection Circuit. The metal oxide varistor (130V/10A) connected across the TIP and the RING lines serves to protect the system from damage caused by telephone line transients, such as those induces by lightning strikes near the telephone line. A bridge rectifier makes sure that the voltage polarity connected across TIP and RING line is the same regardless of which way around the phone wires are connected. As the phone remote system is connected to the telephone line, it must be isolated from high voltages that may occur on it. There are two connections to the line:

- The ring detect interface is isolated with an opto-isolator.
- The DTMF and sound generation interface isolation is done by a transformer.

Ring Detection Circuit. The ring detection circuit, as shown in Figure 2, is the main interface between the phone line and the system. When `PhoneTech` is in on-hook condition, the ring detection circuit is connected to the telephone line. Capacitor C1 is used to block DC to pass through opto-isolator, resistor R1 is used to limit the current passing through opto-isolator LED and the reverse

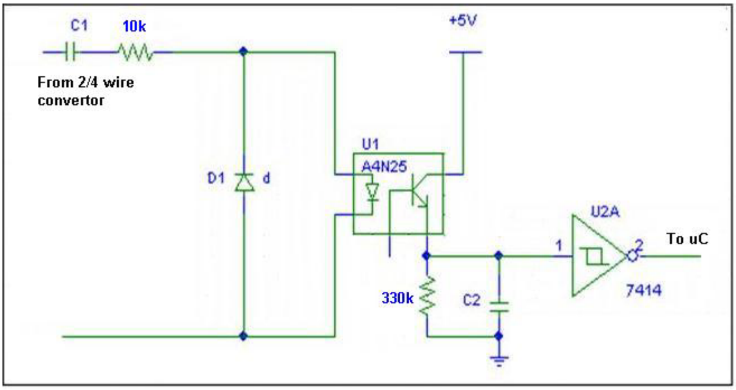


Fig. 2. Ring detection circuit

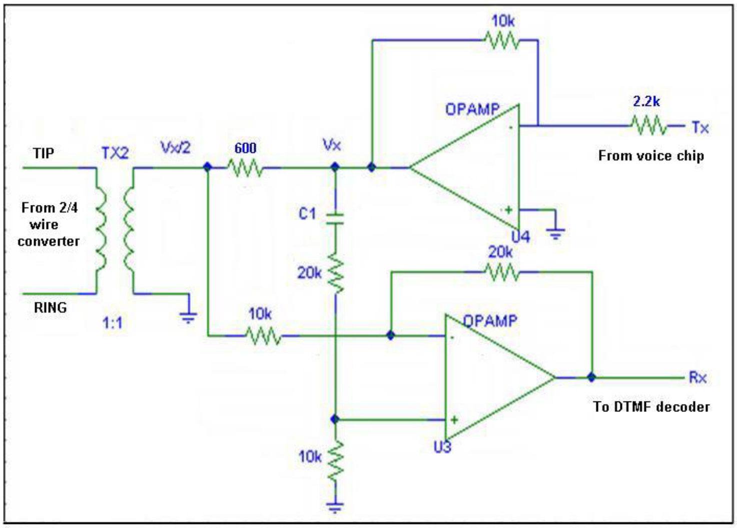


Fig. 3. On/Off-hook circuit

connected diode, D1 which is in parallel with opto-isolator LED, is used to prevent negative voltages from damaging the LED in opto-isolator. The bridge rectifier turns the ringing signal into a pulsating direct voltage which is fed into the opto-isolator.

ON/OFF-Hook Circuit. In normal telephone set, when the user lift up the handset to make call or to answer call, the hook switch is closed and a direct current pass through the telephone line and form a current loop with the Central office (CO). This is the method used to tell the CO that the CO that the telephone set is in off-hook condition. Thus, a single pole double throw (SPDT) relay switch is used to imitate this condition, as shown in Figure 3. An inverting Schmitt trigger which is controlled by the microcontroller is used to drive the LED of the opto-isolator. When the signal applied to this inverting Schmitt trigger is high, it drives the LED ON. Therefore, the phototransistor will start to conduct allowing a direct current to flow through the relay winding and a 1K ohm resistor in series. By this means, the loop current is utilized to energize the relay switch and connect the telephone line to the hybrid circuit, while maintaining as a direct current loop signal to imitate the action of going off-hook. The free wheeling diode connected in parallel with the relay winding provide a return path for the induced current in the winding every time the transistor is switched off and protect the transistor from inductive kick.

3.2 Two-Wire to Four-Wire Converter

Telephone line is full duplex medium. In order to send and receive audio through the pair one must use a two-wire to four-wire converter as depicted in Figure 4,

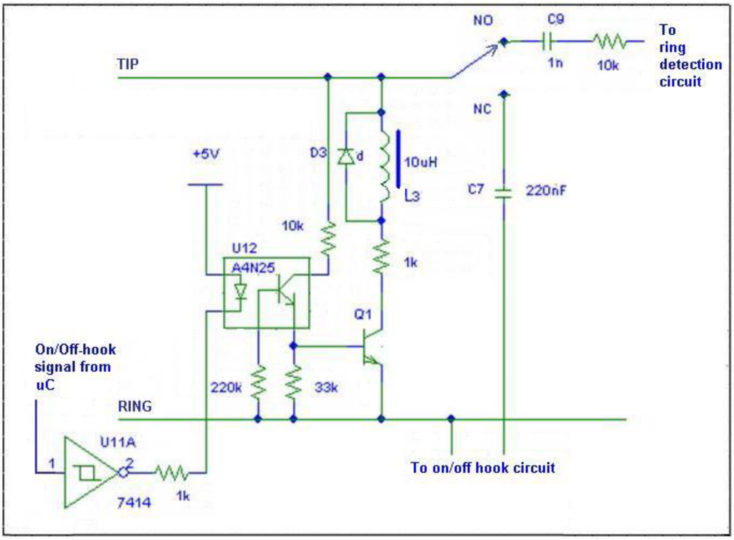


Fig. 4. 2/4 wire converter circuit

which converts the pair into separate transmit and receive audio paths. This hybrid circuit makes it possible to transmit two channels of information in opposite directions on a single pair of wires. The transmission path, Tx is used for audio signal transmission like voice message while the receiving path, Rx is used for receiving DTMF signals which will be fed to the DTMF decoder. The hybrid circuit must provide at least 1500 volt isolation and surge suppression from lightning strikes of the output of the hybrids connected to some other equipment.

3.3 Microcontroller

The microcontroller used in the system is Atmel AT89S51. It is an 8-bit microcontroller. The job of the microcontroller in this system is to control the functions of the other components in the system. It controls the system by performing the following functions at specified time: Detecting ring detection circuit output, switching on/off the On/Off-hook circuit, playing/disabling the voice chip, interpreting the output of DTMF decoder, switching the controlled appliance On/Off, and communicating serially with PC.

3.4 DTMF Decoder

The DTMF decoder decodes DTMF signals and represents them in a sequence of four bits. That means the system receives the DTMF signals as sinusoidal signals and converts them to binary numbers. The DTMF decoder that is used in this system is MT8870 from MITEL. The DTMF signal is decoded and the resulting data is latched in the output register. The decoded data (four bits) is presented at pins D0, D1, D2, and D3 of DTMF decoder.

3.5 Voice Chip

Voice chip is a semiconductor memory chip on which audio can be recorded and be played back just like a typical household recorder. This new technology allows recording of analog audio directly into semiconductor memory without A/D conversion and play it back without D/A conversion. Once the call is picked up, the system offers the user a voice menu of many choices for activating and deactivating appliances at home. The voice menu is recorded on a semiconductor chip. This chip plays the voice menu or other voice messages whenever it is told by the microcontroller to do so. This voice chip is capable of record and playback a message with maximum length of 120 seconds. In the system, voice chip plays the recorded message to the caller for different scenarios. These messages will be recorded one time and then the chip will be only used in the playback mode.

3.6 Infrared Transmitter and Receiver

Infrared (IR) is used to remotely control the home appliances in `PhoneTech` [6]. This infrared remote control transmitter uses 38 kHz to transmit information. Infrared light emitted by IR Diodes is pulsated at 38 thousand times per second. When transmitting it represents logic level "1" and when silence it represents logic level "0". The transmitter uses 555 Timer IC to generate 38 kHz which has to be adjusted using the 10K preset. The duty cycle of the IR beam is about 10% which allows more current to pass through the LEDS thus achieving a longer range. The receiver uses a Sharp GP1u521R IR module. When the IR beam from the transmitter falls on the IR module, the output is pulled to low which activates the relay and deactivated when the beam is obstructed. The relay contacts can be used to switch on/off alarms, lights etc. In this project, a LED is used to imitate the mechanism of switching on/off of a relay.

3.7 Auto-Dialing

If the controlled appliance is a house alarm, this feature may allow the user to set telephone numbers in the PC for emergency call purposes. The number can be the user's cell phone number or any number that the user wants to call whenever there is an emergency occurs. When there is an emergency, the house alarm will be activated. Thus, it triggers the system to make a call automatically to the pre-set number and the voice prompting feature will be on to tell the called party what is happening. The message reported by the telephone is pre-recorded by the user according to the type of emergency. Since a GSM mobile phone is used as a modem to send SMS in this project, it can be utilized to make a call too. By incorporating Telephony API (TAPI) applications in the terminal program written, the program can originate a call through the GSM mobile phone and access to the telephone lines. Telephony API is a single set of functions that can be used to access all aspects of telephony services within the Windows operating system. Auto-dialing is done via an IrDA USB transceiver cable connecting to the GSM phone. The software on the PC will manage the phone number (or a list of phone numbers) and perform auto-dialing through the GSM mobile phone.

4 Concluding Remarks

In this paper, we have presented a new smart home telephone system, `PhoneTech` which was designed to overcome the limitation of distance faced by normal remote control. This system is simple and is based on the telephone function that most house owners are familiar with. `PhoneTech` has the features of automatic off-hook, security password, voice prompting, infrared remote control, system-PC interfacing, SMS notification and auto-dialing during emergency. The system is based on an 8-bit microcontroller. This low cost system will be the every household device in future. The application of this remote control system will grow as the number of phones and particularly cell phone users grow around the world. This system can be further developed to provide better home automation and data communications.

References

1. S. Schneider, J. Swanson and Peng-Yung Woo. Remote Telephone Control System. In *IEEE Transaction on Consumer Electronics*, vol. 43, no. 2, pp. 103-111, 1997.
2. H. Hissen and Zeebar. DTMF Tones. Available at http://www.dialabc.com/sound/dtmf.html, 2004.
3. B. Koyuncu. PC Remote Control of Appliances By using Telephone Lines. In *IEEE Transaction on Consumer Electronics*, vol. 41, no. 1, pp. 201-209, 1995.
4. Mohsen Banan. Computer Telephone Interfacing. *Electrical Engineering, University of Washington*, 1982.
5. P. Horowitz and W. Hill. The Art of Electronics. *Cambridge University Press*, 1989.
6. J. Iovine. DTMF IR Remote Control System. In *Nuts & Volts*, vol. 15, no.6, 1995.

SPAD: A Session Pattern Anomaly Detector for Pre-alerting Intrusions in Home Network

Soo-Jin Park, Young-Shin Park, Yong-Rak Choi, and Sukhoon Kang

Department of Computer Engineering, Daejeon University, Daejeon, Korea
kokiliko@hanmail.net,
good4u@zeus.dju.ac.kr,
{yrchoi, shkang}@dju.ac.kr

Abstract. In order to prevent the intrusion in network-based information systems effectively, it is necessary to detect the early sign in advance of intrusion. This sort of pre-alerting approach may be classified as an active prevention, since detecting the various forms of hackers' intrusion trials to know the vulnerability of systems is not missed and early cross-checked. The existing network-based anomaly detection algorithms that cope with port-scanning and the network vulnerability scans have some weakness in *slow scans* and *coordinated scans*. Therefore, a new concept of pre-alerting algorithm is especially attractive to detect effectively the various forms of abnormal accesses for the trial of intrusion regardless of the intrusion methods. In this paper, we propose a *session pattern anomaly detector* (SPAD) which detects the abnormal service patterns by comparing them with the ordinary normal service patterns.

1 Introduction

The network-based anomaly detection model for the trials of intrusion that make use of port-scanning and search for the vulnerability of network can be referred to Scanlogd as a kind of attacking port scan detection tools and SPADE (Statistical Packet Anomaly Detection Engine) that made in a type of plug-in for Snort [1,2,3,4]. However, they cannot detect the new patterns of the intrusion such as the trial of intrusion. Scanlogd regards the requests of connection as a trial of intrusion, only if the number of times for the requests during the specified interval is more than the predefined threshold. Therefore, if the requests of connection would be generated slowly than the threshold, such slow scanning can not be detected. Also, if several hosts scan ports simultaneously, then such coordinated port-scanning can not be detected.

The algorithm of SPADE has a weak point in that it can not detect the port-scanning which aims for the frequently accessed ports. Since, after it records the frequency of access for all ports of each host, it regards the access for the ports which were not accessed frequently before as the trial of intrusion. Therefore, it can not detect the scan for the ports which were accessed frequently before. For that reason, since the currently well-known Scanlogd and SPADE have restriction in the detectable type of the trial of intrusion, the new concept of algorithm is needed for detecting effectively the various types of abnormal accesses, not constrained to the type of the trial of intrusion.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 432 – 439, 2006.

This paper presents the new concept of pre-alerting intrusion algorithm for SPAD (Session Pattern Anomaly Detector) that regards the session of network service, of which pattern has different than usual, as a trial of intrusion.

2 Session Pattern Anomaly Detector: SPAD

2.1 Notion of SPAD

Regardless of the intension of intrusion, all the packets of users are represented by the service. The session of a service is defined as the packets in which client and server exchange each other during the service, and it is classified into client's session and server's session according to whether the source of the packet is client or server. Since the client and server that participate in the service follow the protocol of the service, there is regularity, that is to say, pattern which is similarly repeated among the sessions of the identical service. The usual sessions follow the normal pattern of the service. But if a session does not follow the normal pattern of the service, it can be regarded as an abnormal access to the port, namely, a trial of intrusion.

Since the trial of intrusion is detected through the pattern of the service protocol, the pattern of the protocol for each service needs to be known. For this, SPAD finds the pattern of the protocol indirectly by learning the traffic of network. The information in packet, that SPAD uses to detect the anomaly, is dIP(destination IP address), dPort(destination Port number) and the size of each packet.

The principle to detect the trial of intrusion by using the pattern of session is as following: An amount of sessions exchanged between clients and servers for normal services are saved statistically for the each dPort of dIP. After that, if a session happens to get out of the statistics, it is regarded as a trial of intrusion. To obtain the patterns of sessions, SPAD uses two kinds of features. The first feature is the length of session that client and server exchange. In other words, the feature is the minimum number of packets which compose the session. If the number of packets exchanged from the beginning to the end of a session of a service is less than that of normal sessions, the session is regarded as an abnormal session and it is judged as a trial of intrusion. The second feature is the common series of packet sizes, which is repeated in all sessions of the identical service. These series are observed at the beginning and at the end of sessions and also observed repeatedly in the middle of the sessions. SPAD adopts the series, which is at the beginning section of session, as the feature and the session whose beginning is different from the normal is regarded and detected as a trial of intrusion.

2.2 Major Components of SPAD

The input and output of SPAD system are traffic and alert respectively. Traffic is the collection of packets which are acquired from network. Alert is the message that informs the trial of intrusion. The structure of system is composed mainly of the three components, namely, Session Classifier, Pattern Extractor and Pattern Comparator. Fig. 1 illustrates the block diagram of the overall structure.

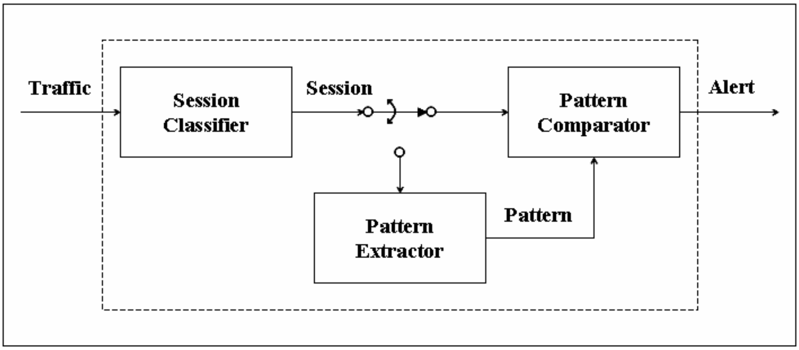


Fig. 1. The Block Diagram of SPAD

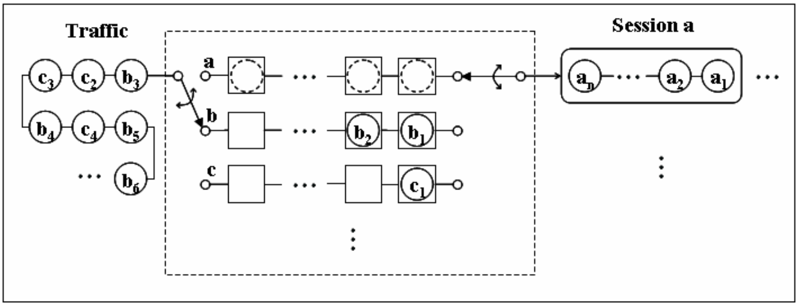


Fig. 2. The Block Diagram of Session Classifier

Fig. 2 is the block diagram of Session Classifier. Session Classifier reads packets from traffic and classify them into the sessions whose source and destination is identical. The source and the destination are the combinations of IP address and port number. For each combination of source and destination, the buffer for storing the packets of the sessions is prepared and, therefore, when the next packet is read from the traffic, it is stored to the corresponding buffer. Then, if all packets in the buffer are stored, all packets of the buffer output as a session. The completed session from the Session Classifier is an input to Pattern Extractor or Pattern Comparator according to the execution mode. The kinds of execution mode are categorized in the form of learning mode and detection mode. In other words, the session output from Session Classifier is an input to Pattern Extractor in learning mode and to Pattern Comparator in detection mode.

The symbols represented in the Fig. 2 are described as following:

- a, b, c : the instances of session (they are different in the combination of source and destination).
- a_i, b_i, c_i : the ith packet of the sessions for a, b and c, respectively.

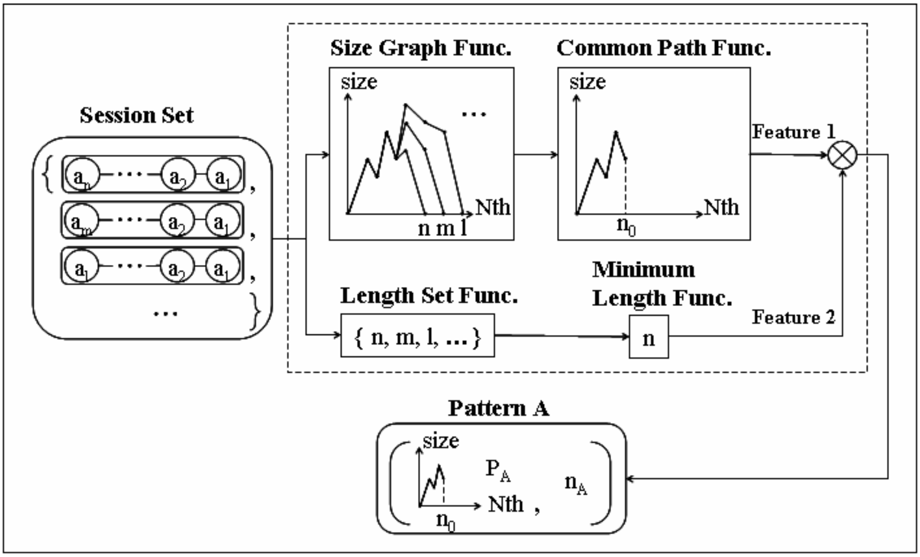


Fig. 3. The Block Diagram of Pattern Extractor

Fig. 3 shows the block diagram of Pattern Extractor. Pattern Extractor collects the sessions which have the identical destination and finds and output the common pattern among them. Therefore, the pattern is extracted for each destination.

The pattern of a service is composed of two features. The first is the common series of packet sizes in the beginning of the service. As it to say, for the data of sessions which have the identical destination, when the sizes of packets of each session are superposed (overlapped) in time series on those of other sessions, the common section of the superposed time series will appear. That common section among the sessions is the first feature for the pattern of the service. The second feature is the minimum size of session during the service, where the size of session is the number of the packets which compose the session.

To obtain the two features, as input, Pattern Extractor acquires a set of sessions that have the identical source and destination and input it both to the Size Graph Function and Length Set Function. Size Graph Function makes a tree for the packets of all the sessions using the packet sizes as nodes. This tree is input to Common Graph Path Function and the common path is found out. This common path of tree becomes the first feature of pattern. Length Set Function records the sizes of all the sessions and output them. Minimum Length Function receives this data and finds and output the minimum size of them. This minimum size of session becomes the second feature of pattern.

The first and second features are paired and output. The symbols represented in the Fig. 3 are described as following:

▪ a, a', a":	the instances of session (they are identical in source and destination)
▪ a_i, a_i', a_i":	the i^th^ packet of the sessions for a, a' and a", respectively
▪ n, m, l:	the length of session for a, a' and a", respectively (total number of packets in a session)

- n_0: the length of common time series of packet sizes for the sessions a, a' and a''
- A: the pattern for the service whose source and destination combination is identical with that of sessions a, a' and a''
- P_A: the first feature of pattern A (the common time series of packet sizes of sessions for the service)
- n_A: the second feature of pattern A (the minimum length of session for the service)
- n_{A0}: the length of P_A
- size: the size of packet (bytes)
- Nth: the index of packet

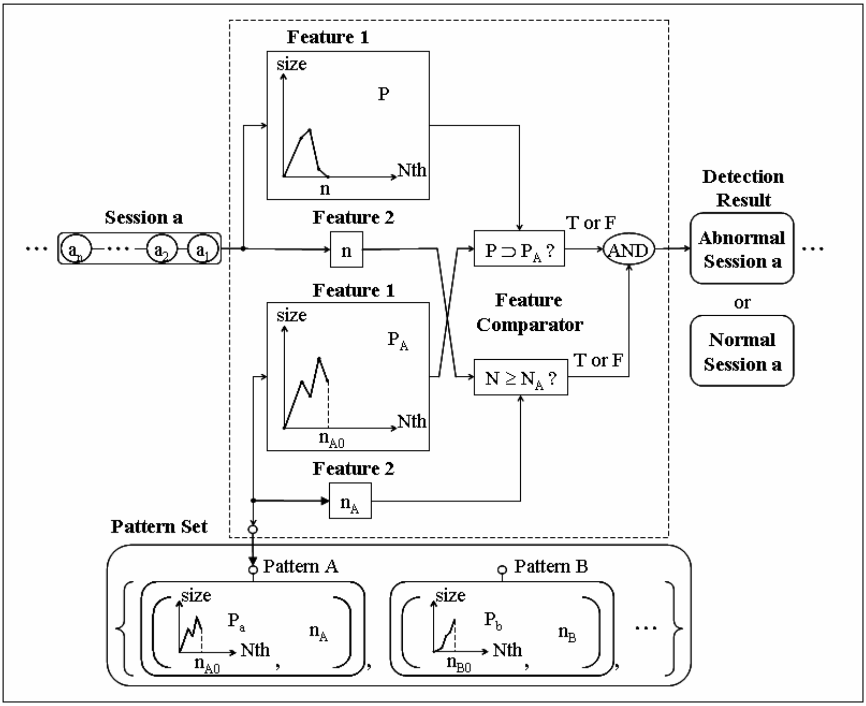


Fig. 4. The Block Diagram of Pattern Comparator

Fig. 4 is the block diagram of Pattern Comparator. Pattern Comparator compares the current input session with the pattern extracted previously for the service. And if the session does not follow the pattern, then, Pattern Comparator regards it as an abnormal session and reports an alarm. Therefore, Pattern Comparator receives the two inputs - the session and the pattern.

From the input session, the first feature and the second feature of it are extracted similarly as the Pattern Extractor does. In other words, the first feature is the time series of packet sizes and the second feature is the length of the session. In comparing the two features session and the pattern, if one of the two features is different from the other, then, Pattern Comparator alerts an alarm for the abnormal session.

The symbols represented in Fig. 4 are as following:

- a: the current input session to be examined
- a_i: the ith packet of the session a
- P: the first feature of the input session (the time series of packet sizes of input session)
- n: the second feature of the input session (the length of input session)
- A, B: the patterns saved before (where, the source and destination combination of the pattern A is identical to that of session a)
- P_A, P_B: the first feature of pattern A and B respectively (the common time series of packet sizes of sessions for the service)
- n_A, n_B: the second feature of pattern A and B respectively (the minimum length of session for the service)
- n_{A0}, n_{B0}: the length of P_A and P_B respectively.
- size: the size of packet (bytes)
- N^{th}: the index of packet

In the implementation of Pattern Extractor, there is a selection on whether the pattern extraction would be continued during the anomaly detection or performed separately. If the former is called online pattern extraction and the later is called offline pattern extraction, the online pattern extraction has the merit that it reflects the recently varied pattern of sessions for the service. But if the trial of intrusion happens while performing the online pattern extraction, it has the defect that it reflects the trial of intrusion on the patterns. The pattern must reflect the normal traffic. Otherwise, the same trial of intrusion would be considered as a normal session. However, the offline pattern extraction is used currently, the method to consider the merits of two alternatives should be devised later.

3 Analysis and Interpretation of Simulation Results

For the reliability of the evaluation of SPAD model, in the evaluation data for intrusion detection, which is made by MIT Lincoln Research Center during DARPA project, the data set of the year 1999 was used to the simulation for the evaluation [7, 8, 9, 10].

The above evaluation data for intrusion detection has 5 weeks traffic of packets and the 1st, 2nd and 3rd week data are normal traffics and the 4th and 5th week data are the traffics including attacks. The evaluation of SPAD used the 1st and 3rd week data for the pattern extraction of services and used the 4th week data for the detection of anomaly. In the 4th and 5th week data, there are several classes of attacks as the Table 1. Among them, since the trial of intrusion corresponds to the Probes, therefore, the evaluation used only the Probes.

Table 1. The kinds of attacks in evaluation data

Class	Types
Denial of Service Attacks	Apache2, arppoison, Back, Crashiis, dosnuke, Land, Mailbomb, SYN Flood, Ping of Death, Process Table, selfping, Smurf, sshprocesstable, Syslogd, tcpreset, Teardrop, Udpstorm
User to Root Attacks	anypw, casesen, Eject, Ffbconfig, Fdformat, Loadmodule, ntfsdos, Perl, Ps, sechole, Xterm, yaga
Remote to Local Attacks	Dictionary, Ftpwrite, Guest, Httptunnel, Imap, Named, ncftp, netbus, netcat, Phf, ppmacro, Sendmail, sshtrojan, Xlock, Xsnoop
Probes	insidesniffer, Ipsweep, ls_domain, Mscan, NTinfoscan, Nmap, queso, resetscan, Saint, Satan

Table 2. The simulation results of evaluation for SPAD

Day of 4th week	n(TP)	n(FP)	n(FN)	n(TN)	R(TP)	R(FP)	R(FN)	R(TN)
1st	7	27	0	3,487	100%	0.8%	0%	99.2%
3rd	11	186	6	3,329	64.7%	5.3%	35.3%	94.7%
4th	3	511	0	3,507	100%	12.7%	0%	87.3%
5th	5	511	0	1,393	100%	26.8%	0%	73.2%
Total	26	1,235	6	11,716	81.3%	9.5%	18.7%	90.5%

The symbols represented in the Table 2 are described as follows:

- n(*): the count of occurrence, R(*): the rate of occurrence
- TP: true positive, FP: false positive, FN: false negative, TN: true negative

For the performance evaluation of SPAD, the simulation was performed and the result of evaluation has been obtained, shown Table 2. The result has been counted per session not per packet since SPAD performs based on session.

According to the analysis of results, the reason why the average of R(TP) is 81.3% is the insufficiency of the two weeks data used to extract the normal pattern of services. Because the reason why the average of R(FP) is 9.5% is SPAD detected what the evaluation data missed to put in the attack list. By using tcpdump command, they can be confirmed to be abnormal accesses to ports. As consequence, SPAD has found them although they were missed in the attack list prepared.

4 Conclusions

To detect the port-scanning and the search of vulnerability for network, which the attacker tries prior to his intrusion, the paper presents SPAD (Session Pattern Anomaly Detector) to detect the abnormal session for trial of intrusion, using the normal pattern of session for services. As the clue of the judgment on the trial of intrusion, SPAD focuses on whether the remote host users really use the services. So SPAD Stores usually the normal pattern of services; when a service pattern differs from the stored, SPAD detects it as a trial of intrusion.

The two features of pattern extracted are: first, the common time series of packet sizes in the beginning of session data exchanged during the service between client and server and, the second, the minimum number of packets for the session of service. For the performance evaluation of SPAD, the simulation was performed by using the “IDS Evaluation Data Set” made by MIT. And the average rates of true positive and the false positive has been obtained as 81.3% and 9.5% respectively. The true positive rate of 81.3% is considered to be caused by the insufficiency of data used to extract the various normal patterns of services because the evaluation data set was as short as 2 weeks.

The SPAD model can detect slow scanning, coordinated scanning and the scan for the ports which were accessed frequently, which are not detected by the existing algorithms, Scandlogd and SPADE.

Acknowledgement

This research was supported by the Program for the Training of Graduate Students in Regional Innovation which was conducted by the Ministry of Commerce, Industry and Energy of the Korean Government.

Reference

1. Solar Designer, “Designing and Attacking Port Scan Detection Tools,” Phrack Magazine Volume 8, Issue 53, July 8, 1998.
2. Fyodor, “The Art of Port Scanning,” Phrack Magazine, Volume 7, Issue 51 September 01, 1997.
3. “Publication of Real-time Network Illegal Scanning Automatic Detection Tool (RTSD),” http://www.certcc.or.kr/
4. http://www.silicondefense.com/software/spice/index.htm
5. Stuart Staniford, James A. Hoagland and Joseph M. Mcalerney, “Practical Automated Detection of Stealthy Portscans,” http://www.silicondefense.com/software/spice/index.htm
6. James A. Hoagland and Stuart Staniford, “Viewing IDS alerts: Lessons from SnortSnarf,” IEEE, 2001.
7. John McHugh, “Testing Intrusion Detection Systems: A Cririque of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory,” ACM Transactions on Information and System Security, Vol. 3, No. 4, pp. 262-294, November 2000. Others
8. http://www.ll.mit.edu/IST/ideval/index.html
9. Attack database, http://www.ll.mit.edu/IST/ideval/docs/docs_index.html
10. Off-Line Simulation Network, http://www.ll.mit.edu/IST/ideval/docs/docs_index.html

Home Gateway with Automated Real-Time Intrusion Detection for Secure Home Networks[⋆]

Hayoung Oh[1], Jiyoung Lim[2], Kijoon Chae[1], and Jungchan Nah[3]

[1] Dept. of Computer Engineering, Ewha Womans University, Seoul 120-750, Korea
{hyoh, kjchae}@ewha.ac.kr

[2] Dept. of Internet and Information, Korean Bible University, Seoul 139-791, Korea
jylim@bible.ac.kr

[3] Protocol Engineering Center, ETRI, Daejeon 305-350, Korea
njc@etri.re.kr

Abstract. Home networks will be widely established in residential areas. Intrusion detection is an important function in the home gateway because various networks try to access to home networks. We propose the home gateway with the automated real time intrusion detection adjustable in home network environment using the clustering methodology and the correlation. Our proposed model showed the reasonable misclassification rates.

1 Introduction

A home network interconnects electronic products and systems, enabling remote access to and control of those products, systems and any available content such as audio, video, or data. Ethernet, IEEE 1394, wireless networks and the power line network could be used as home network technologies. Public Switched Telecommunication Networks(PSTN), Integrated Services Digital Networks(ISDN), broadcasting networks(satellite, terrestrial and CATV) and wireless access and the Internet could be used as home network access networks [1].

The home gateway performs a role of an intermediary between access networks and the home network as shown Fig. 1. The first function is providing the capability of remote control of the connected appliances on the home network from various access networks. Second is the security function against various intrusions. Others are GUI for public networks, auto configuration through the address/protocol translation, the media translation capabilities and etc [2]. In this paper, we focused on the security features among all functions of home gateways because intruders always seek to gain control of your computer and having control of your home computer system enables those to easily control your networked home devices or even launch attacks on other systems.

Many security products generally tend to have been designed based on requirements for the industry than those of the home. There are lack integrated

[⋆] This work was supported by University IT Research Center(ITRC) Project and Electronics and Telecommunication Research Institute(ETRI) Project.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 440–447, 2006.

security solutions for the home network to protect it from wired and wireless attacks. Therefore, we will propose the firewall to detect intrusions in home network environments.

There are three strategies in the intrusion detection field. Misuse detection is based on the pre-built set of intrusion scenarios, which could detect the already known attacks with high accuracy. A drawback of misuse system is that it would fail to detect the unknown attacks. Anomaly detection is based on the hypothesis that the malicious behavior should be different from the normal ones. The normal behaviors could be characterized by a model and the anomalies could be identified by the deviations from this model. It offers the advantage of finding the unknown attacks. Specification based detection focuses on abstracting the normal behaviors of critical objects. Intrusions which usually cause objects to behave in an incorrect manner can be detected without exact knowledge about them. It could generate fewer false alarms compared with those of anomaly detection. But it will cost more efforts on building the specifications for specific objects [3].

Anomaly detection could be suitable to detect intrusion in the home network because there are no decision rule maker(or analyst) in the home network in contrast to the industry but various unknown attacks same as the industry. Clustering among anomaly detection methodologies groups similar data together under no supervision, eases the tasks of labeling by experts. Self-Organized Map(SOM) is one of clustering algorithms and is a data visualization technique which reduces the dimensions of data through the use of self-organized neural networks [4].

Gonzales et al. compared the Neuro-Immune and SOM in terms of the classification rates. They showed their classification rates were similar and very correct [5]. Jirapummin et al. proposed the intrusion detection mechanism using SOM for clustering and RPROP (Resilient Propagating Neural Network) for labeling [6]. However, RPROP cannot accommodate a new untrained attack and its performance depends on the number of neurons in the map. SOM is useful for classification of the known and unknown attacks and the normal but it does not enable to label them. We also use the correlation methodology for labeling and analysis on traffic features while using SOM for classification.

We propose the intrusion detection system for home gateway in home network environments to satisfy the above requirements. We use the clustering methodology and the correlation to make the intrusion detection classify even unknown attacks and alarm them to deal with them as soon as they intrude.

In this paper, we describe the proposed intrusion detection in section 2 and its experimental results in section 3. Finally, we concluded in section 4.

2 The Proposed Home Gateway with Intrusion Detection

Fig. 2 shows our system with three steps as follows: Training, Labeling and Detection & Training.

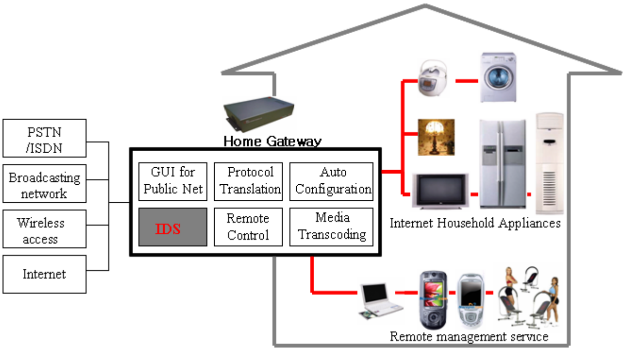


Fig. 1. A Home Gateway Architecture

2.1 Step 1: Training

The accurate training requires the modification of data because the values of traffic features have the various ranges. Our mechanism has the preprocessing stage and the normalization stage before training.

Preprocessing. It transforms the TCPdump data into the numeric data because SOM resolves only numerical data but the TCPdump data has some features whose data types are not numeric.

Normalization. SOM makes the maps for each feature respectively and then construct the U-matrix(unified matrix) based on the all feature maps. Every feature has such various ranges. For example, the values of the feature *src_byte* are in the range between 0 and 2194619, the feature *duration*'s values are in the range between 0 and 42448 and some feature's values are in the range between 0 and 1. Some wide range features such as *src_byte* and *duration* affect the U-matrix construction much more than any other features. Thus normalization is needed to make U-matrix reflecting all features fairly. Our normalization makes the minimum values of every feature 0 and their maximum values 1. The normalization equation used in this paper is as follows:

$$N_{i(x)} = (i(x) - V_{min}(x))/(V_{max}(x) - V_{min}(x)),$$

where x means one of features, i(x) is the original data value of the feature $x(V_{min}(x) \leq i(x) \leq V_{max}(x))$, $V_{min}(x)$ is the minimum value of the feature x and $V_{max}(x)$ is the maximum value of the feature x.

Training. SOM is not only the clustering model using the neural network method but also the unsupervised learning model. The unsupervised learning automatically categorizes the varieties of input presented during training and enable to match in which neuron new inputs concern while the supervised learning determine which output one of many possible input values matches for.

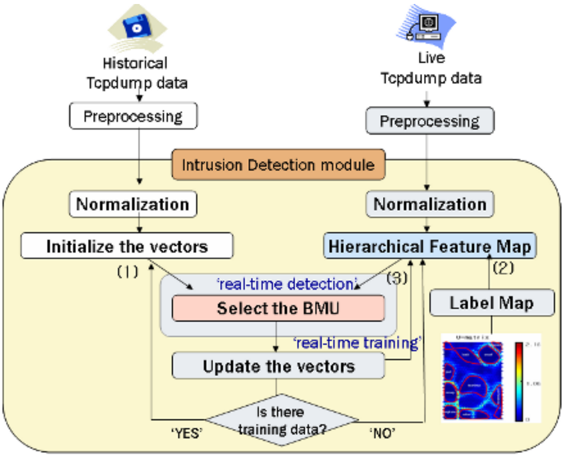


Fig. 2. Process of Our Intrusion Detection System in Home Gateway

2.2 Step 2: Labeling

It is difficult deciding which SOM cluster is normal or abnormal and which attacks the abnormal input is because feature maps and U-matrix give no information about input data. To solve this problem, correlations between features on each attack are analyzed. Pearson correlation coefficient equation used to analyze them is as follows [7]:

$$r_p = \frac{\sum(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum(x_i - \overline{x})^2}\sqrt{\sum(y_i - \overline{y})^2}}$$

where $\overline{x} = \frac{1}{n}\sum_{i=1}^{n} x_i\,,\;\; \overline{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ and $(x_i, y_i) \in \{(x_1, y_1), \cdot\cdot\cdot, (x_n, y_n)\}$

Table 1. shows the correlation results between features on each attack. For instance, the attack Pod has the 3 feature-pair correlations such as (21+23) and 3-(5+8). (21+23) means the positive correlation between feature no. 21(the percentage of connections to the same service at destination host) and 23(the percentage of connections to the same source ports at destination host). 3-(5+8) means the minus correlations between feature no. 3(the network service types on the destination) and 5(the number of data bytes from source to destination) and between 3 and 8(the number of ”wrong” fragments) and the positive correlation between 5 and 8. Fig. 3 shows U-matrix and Feature Maps only for the attack Pod. U-matrix shows results analyzing which cluster in U-matrix is abnormal or normal through the correlation results between features on each attack in Table 1. For instance, the most left and center cluster means the attack Pod. The largest part not circled means normal clusters. Feature Maps only for the attack Pod where the color distribution and shape of the circled feature map for the feature no. 5 are very similar to those of 8 but are opposed to those of 3 and the maps for feature no. 21 and 23 are much alike in the color distribution and shape. Numbering features are based on [8].

Table 1. Correlations between features on each Attack

Normal	$= (10{+}11){\cap}(12{+}13){\cap}(14{+}15){\cap}(16{-}17){\cap}(20{+}21)$
Neptune	$= (11 + 16)$
Smurf	$= (10 + 11) \cap (19 + 20) \cap \{(21 + 23) - 22\}$
Teardrop	$= (20 + 21 + 23)$
Back	$= (12 + 13) \cap (25 + 26) \cap (27 + 28)$
Pod	$= (21 + 23) \cap \{3 - (5 + 8)\}$
Ipsweep	$= (3 + 19 + 22) - (21 + 23)$
Nmap	$= \{(4 + 12 + 13 + 26) - 21\}$
Portsweep	$= (22 + 23 + 27) \cap (15 + 28)$
Satan	$= (10 + 22) \cap (14 - 23) \cap (15 + 28)$

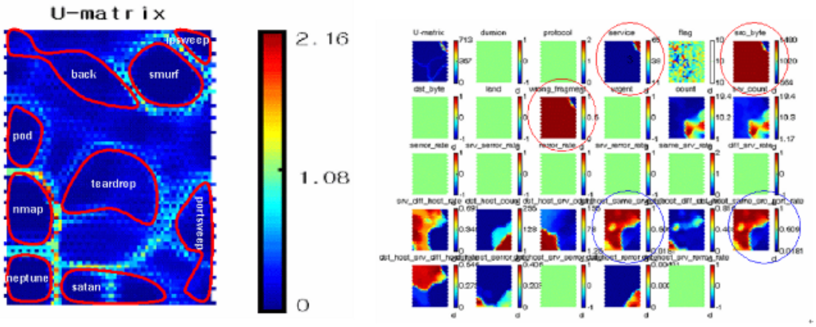


Fig. 3. U-matrix and Feature Maps for Pod attack

2.3 Step 3: Detection and Training

The proposed real time intrusion detection is achieved by finding BMU (the best matching unit) with the smallest Euclidean Distance measured between input data and the map unit. If BMU is a cluster in the normal cluster set, the input is normal. Otherwise, it is abnormal. Our detection process is as follows:

Algorithm 1. Intrusion Detection Algorithm using Euclidean Distance

```
1. $BMU = arg_{min} \parallel (x(n) - w_j(n) \parallel$
   where $x(n)$ is the input vector being presented at time $n$ and $w_j(n)$ is the
   weight vector for all nodes in the network.
2. If $BMU \in$ the set of normal clusters,
   then $x(n) = normal$
   else $x(n) = abnormal$
```

After the detection process, weights of BMU and its neighbors are updated based on the equation as follows:

$$w_i(n+1) = w_i(n) + h_{ci}(n)[x(n) - w_i(n)]$$

3 Experimental Results

We use a part of DARPA 1998 Intrusion Detection Evaluation data set used in the 3rd International Knowledge Discovery and Data Mining Tools Competition in 1999(KDD Cup 1999) [8]. Although many flaws exist in the KDD dataset as discussed in [9], to our knowledge the KDD Cup 1999 contains relatively many attack types in the sense of public evaluation platforms for intrusion detection. It includes some 7 million TCP connection records and consists of the labeled training data with about 5 million connections (KDD-TND) and the test data with 2 million connections (KDD-TD). We use two subsets to compare the detection rates of Decision Tree and Neural Network of supervised learning techniques with those of our proposed real-time intrusion detection system based on SOM and feature correlations. We consider Back, Neptune, Pod, Smurf, teardrop, Ipsweep, Nmap, Portsweep and Satan as attacks. 70% of KDD-TND is used for training and 30% of KDD-TND and KDD-TD are used for test of anomaly detection and misuse detection.

To evaluate the performance of misuse detection, the system should train the data including normal and all considering attacks and then catches intrusions in terms of the characteristics of known attacks. We train the mixture of normal and attacks and build and label the U-matrix using SOM Toolbox based on Matlab to evaluate the performance of misuse detection in our system [10]. Then we collect the BMU results for test data in our system. To evaluate the performance of misuse detection in the supervised learning technique, we also use same training data in our system as input data to Decision Tree and Neural Network of SAS Enterprise Miner [11].

Table 2 shows the misclassification rates in terms of the misuse detection. Since the misclassification means that our system selects the wrong BMU and Decision Tree and Neural Network system does not detect accurately, the lower rate is the better. The rate of 30% KDD-TND is lightly lower than that of KDD-TD because 30% of KDD-TND is familiar with 70% of KDD-TND. U-matrix in our systems and results of Decision Tree and Neural Network more adjustable to the characteristics of 30% of KDD-TND. We also knew that our system detect more accurately than that of the supervised learning. Anomaly detection means that the intrusion system catches intrusions of the new attack not trained. The supervised learning technique and our system trained respectively for 70% of KDD-TND without only one attack among 9 attacks. In the case of the anomaly detection for the attack Back, each system is trained for the mixture of normal and attacks without Back and test whether each system decide that attack Back is abnormal. Table 3 shows the misclassification rates of 9 attacks respectively for 30% KDD-TND and those rates for KDD-TD. Our system showed that the

Table 2. Misclassification Rates as the Misuse Detection

Data	Supervised System	Unsupervised System
30% of KDD-TND	0.25	0.03
KDD-TD	0.59	0.05

Table 3. Misclassification Rates as the Anomaly Detection

Data	Attack	Supervised System	Unsupervised System
30% of KDD-TND	Back	0.087	0.000
	Neptune	0.150	0.050
	Pod	0.264	0.054
	Smurf	0.162	0.092
	Teardrop	0.924	0.045
	Ipsweep	0.093	0.057
	Nmap	0.735	0.035
	Portsweep	0.505	0.058
	Satan	0.246	0.049
KDD-TD	Back	0.989	0.002
	Neptune	0.361	0.068
	Pod	0.694	0.059
	Smurf	0.369	0.099
	Teardrop	0.997	0.071
	Ipsweep	0.389	0.067
	Nmap	0.908	0.049
	Portsweep	0.826	0.069
	Satan	0.744	0.058

Table 4. Modeling Time and Detection Time

	Supervised	Unsupervised
Modeling Time	15.80s	12.40s
Detection Time	1.90s	0.50s

anomaly detection of the attack Back is perfect. Its characteristics are clear while those of the attack Smurf are a little dim. Also Our system detected more accurately than that of the supervised learning.

To evaluate possibility of real time intrusion detection of the supervised learning technique and our system, we experimented the modeling time and the detection time of each system. The modeling time means the average time taking to train the collected input data and the detection time means the average time taking to decide whether the new input is normal or abnormal after training. Table 4 shows the detection time of our system is 0.5 seconds in whole experiments while that of the supervised learning system is 1.9 seconds, which means our system can be called the real-time intrusions detection system. We also knew that our system takes little time to model intrusion system than the supervised learning system.

4 Conclusion

In this paper, we proposed the home gateway with automated real-time intrusion detection for secure home networks. We use the clustering methodology and the correlation to make the intrusion detection classify even unknown attacks and alarm them to deal with them as soon as they intrude. We used KDD Cup 1999 training and testing dataset and SOM, one of unsupervised learning data mining and clustering mechanisms to validate classification ability of the misuse detection and anomaly detection. Our system yielded the reasonable misclassification rates through several experiments than those of the previous suggested supervised learning system. We got the characteristics of each attack whose unclassifiable features look to have no relations among themselves. Our feature correlation results got for labeling can be used in other intrusion detection systems using other technologies even if a new attack happens because our correlation information are adjusted as time grows. We will analyze attacks with unclear characteristics such as Smurf, extract important features to reduce the process overhead and then make our system more accuracy.

References

1. Saito, T., Tomoda, I., Takabatake, Y., Arni, J., Teramoto, K., "Home gateway architecture and its implementation," IEEE Transactions on Consumer Electronics, p.1161 - 1166 , Nov. 2000.
2. Zhefan Jiang, Sangok Kim, Kanghee Lee, Hyunchul Bae, Sangwook Kim, "Security Service Framework for Home Network," The Fourth Annual ACIS International Conference on Computer and Information Science (ICIS05), p.233-238, Jul. 2005.
3. Shu-Yuan Jin, Yeung, D.S., "DDoS detection based on feature space modeling," 2004 International Conference on Machine Learning and Cybernetics, p.4210-4215, Aug. 2004.
4. Tom Germano, "Self Organizing Maps",
Available in http://davis.wpi.edu/ matt/courses/soms/.
5. Fabio Gonzalez, Dipanker Dasgupta, "Neuro-Immune and Self-Organizing Map Approaches to Anomaly Detection: A comparison", ICARIS, 2002.
6. Chaivat Jirapummin, Naruemon Wattanapongsakorn, Prasert Kanthamanon, "Hybrid Neural Networks for Intrusion Detection System", King Mongkut's University of Technology Thonburi, 2001.
7. Pearson Correlation Coefficient,
Available in http://www.indstate.edu/nurs/mary/N322/pearsonr.html/.
8. KDD Cup 1999 Data,
Available in http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.
9. H. Gunes Kayacik, A. Nur Zuncir-Heywood, Malcolm I. Heywood, "On the Capability of an SOM based Intrusion Detection System", International Joint Conference on Neural Networks, 2003.
10. Juha Vesanto, John Himberg, Esa Alhoniemi, and Juha Parhankangas, "SOM Toolbox for Matlab 5", SOM Toolbox Team, Helsinki University of Technology, 2000.
11. Mihui Kim, Hyunjung Na, Kijoon Chae, Hyochan Bang, Jungchan Na, "A Combined Data Mining Approach for DDos Attack Detection," ICOIN 2004, LNCS 3090, p.943-950, Feb. 2004.

The Performance Analysis of UWB System for the HD Multimedia Communication in a Home Network*

Chul-Yong Uhm, Su-Nam Kim, Kyeong-Hoon Jung, Dong-Wook Kang,
and Ki-Doo Kim

School of Electrical Engineering, Kookmin University,
861-1, Chongnung-dong, Songbuk-gu, Seoul, Korea
{cyuhm, snkim, khjung, dwkang, kdk}@kookmin.ac.kr
http://dsp.kookmin.ac.kr

Abstract. In this paper, we propose the UWB system as a wireless HD multimedia transmission technology in a home network. We examine the piconet of UWB system and analyze the performance by using the development kit in a real office and empty space environments.

1 Introduction

As the digital multimedia appliacaitons, such as digital broadcasting, DVD(Digital Versatile Disk) and home theater, become familiar services, it is required to transmit and recieve the high quality video in a home network. Digital broadcasting provides the HD(High Definition) video which has five or six times higher resolution than analog broadcasting. Thus it needs bandwidth of the above 20Mbps for the transmission of MPEG transport stream. Also the DV(Digital Video), which is the data format of a digital camcorder, requires more than 25Mbps for a real time transmission. However, the technologies of up-to-date home network are mainly focused for home automation and they provide relatively low data rate which is not sufficient for HD multimedia communication. We consider the UWB(Ultra Wideband) system can be one of the candidates for that purpose in a home network, because it has the highest data rate and low power consumption among various wireless communication systems. Moreover, UWB system has additional advantages for home environment, since the piconet can be successfully integrated with it. In this paper, we investigate the current state of UWB technology including network structure to communicate the HD multimedia data. And we analyze the performance using XSUWBWDK, the UWB development kit of Freescale^{TM}.

2 UWB System

The UWB system is being paid attention as a physical layer of the WPAN(Wireless Personal Network) [1], because it has many desirable features, such as high data rate, low power consumption, and small size. The UWB system, which is originally

* This work was supported by research program 2005 of Kookmin University in Korea.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 448 – 455, 2006.

developed for military applications, was open to the public, as UWB 1st Report & Order was accepted by FCC(Federal Communications Commission)[2]. This R&O classified UWB system into three fields: imaging systems, communication and measurement systems, and vehicular radar systems. The IEEE 802.15 WG(Working Group) started the IEEE 802.15.3a SG(Study Group) in November 2001, in order to apply the UWB system for the high speed wireless data transmission. And the TG(Task Group)3a began to take action in January 2003[3]. According to the definition of FCC, the UWB signal has the bandwidth greater than 20% of center frequency or greater than 500MHz. Thus the UWB bandwidth can be represented by (1) where f_u and f_l are the upper and lower 10dB down point, respectively.

$$BW = \frac{f_u - f_l}{\left(f_u + f_l\right)/2} \geq 0.2 \qquad (1)$$

The standardization process is currently on going, but the schedule is delayed due to the disagreement between the Multi-Band OFDM[4] of MBOA(Multi Band OFDM Alliance) proposed by TI and CDMA-based DS-UWB[5] proposed by Motorola.

2.1 The Piconet

The piconet is composed of a PNC(Piconet Coordinator) and several DEV(Device)s as shown in Fig. 1. One of DEV can be operated as PNC and the communication begins by sending a beacon from this PNC to one of DEVs. The transmission coverage of a piconet is normally about 10m.

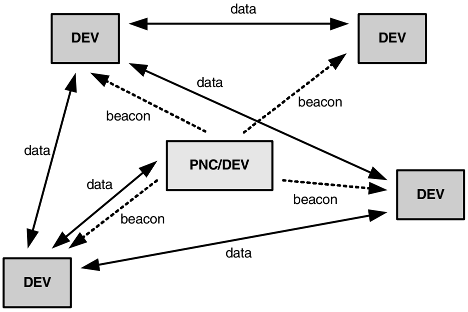


Fig. 1. The topology of piconet

A data format is divided into several superframes in IEEE 802.15.3 MAC(Media Access Control) protocol and a superframe is composed of three blocks as shown in Fig. 2. The first one is beacon and the second is CAP(Contention Access Period) which performs the random access control. The third is CTAP(Channel Time Allocation Period) which stores the payload data[6].

The control data for data transmission is included in a beacon. The CAP is used for random access based on CSMA/CA(Carrier Sensing Multiple Access/Collision Avoidance) method. The CAP is not mandatory. When CAP is omitted, the

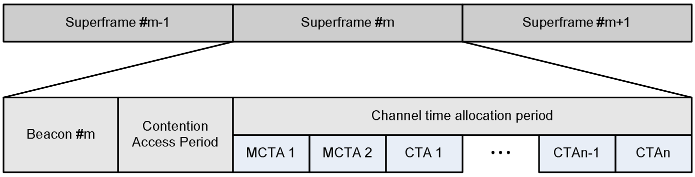


Fig. 2. The superframe of IEEE 802.15.3

MCTA(Management CTA) in CTAP can be used for random access by using the slotted Aloha method. Each DEV competes to get a right to communicate with PNC. After this process, it is possible to transfer payload data. A synchronous data such as streaming video can be transmitted in allocated CTA in CTAP. Also an asynchronous data can be transmitted with a variable bandwidth which is adaptively allocated according to the available data rate.

2.2 UWB System

There are two types of popular wireless networking technologies, WLAN(IEEE 802.11a/g) and UWB, for HD multimedia communication since the data rate of these two technologies are more than 30Mbps. Table 1 shows the differences between WLAN(IEEE 802.11a/g) and UWB. WLAN is a popular wireless technology but the maximum data rate is only 54Mbps and the service area is limited within the coverage of 5m radius at that rate. That mean the WLAN can transmit just a single HD quality video and the data rate of which is not sufficient to transfer a lot of HD video data. Also, AP(Access Point) overload is likely to take place in the ad-hoc network of WLAN since it supports only infrastructure network due to the problems such as power consumption, security and QoS(Quality of Service). On the contrary, UWB can supply several HD videos because it supports the data rate of 100Mbps within 10m and its topology is a piconet network. One of the troubles of UWB system is the relatively small coverage. Thus backbone network or repeater needs to be incorporated with the current UWB system to construct a home network.

Table 1. Comparison of WLAN(IEEE 802.11a/g) and UWB

	Data rate	Coverage	Topology	Disadvantage
IEEE 802.11a/g	~54Mbps	Upto 50m	Infrastructure	ISM or UNII band
UWB	~1.2Gbps	Upto 10m	Piconet	Not standardized

The bandwidth allocated for UWB system is 7.5GHz. And the DS(Direct Sequence)-UWB system proposed by Motorola use two sub-band in order to coexist with WLAN(IEEE 802.11a)[6] which uses UNII(Unlicensed National Information Infrastructure) frequency band. Multi-band OFDM and DS-UWB systems are compared in Table 2.

Table 2. Comparison of Multi-band OFDM and DS-UWB

	Multi-Band UWB	DS-UWB
The number of bands	Multi-band(13)	Dual band
Modulation	OFDM/QPSK	CDMA(M-BOK)/PSK
FEC	Convolutional Code	Convolutional Code
Data rate	55~480 Mbps	25.8 Mbps ~ 1.2Gbps
The number of piconets	4	6
Complexity	FFT/IFFT structure	Rake receiver
Feature	Robust to multipath, Peak-to-Average power ratio problems	Robustness to interferences

If we let *m*(*t*) as the pulse signal after RRC(Root Raised Cosine) filtering, then the RF modulated signal becomes *m*(*t*)*cos*(ω_c *t*) and its Fourier transform is given as (2)

$$\int_{-\infty}^{\infty} m(t)\cos(\omega_c t)\, e^{-j\omega t} dt = \frac{1}{2}\left[M(\omega+\omega_c) + M(\omega-\omega_c)\right] \tag{2}$$

The *M*(ω) is the Fourier transform of pulse signal *m*(*t*), and the carrier frequency ω_c is 2π×4.104GHz. The bandwidth of baseband signal *m*(*t*) is 684MHz and the bandwidth of RF signal is 1.367GHz.

3 The Development Kit: XSUWBWDK

We use the UWB development kit, XSUWBWDK(Xtreme Spectrum Ultrawide Band Wireless Development Kit) of Freescale™ [7], to analyze the performance of DS-UWB system for the HD multimedia communication . Features of XSUWBWDK are represented as

- IEEE 802.15.3 MAC protocol
- Software applications programming interface
- Ability to transfer media-rich streams via a 1394 interface
- Test and configuration utilities that streamline radio evaluation
- Bi-phase encoding that supports data rates to 114Mbps
- Selectable forward error correction values: 1, 3/4, and 1/2
- Selectable data rates: 28.5, 57, and 114Mbps

XSUWBWDK can be classified to four parts: MAC part, Power part, 1394 interface part and PHY part. The PHY part can be divided into two parts as digital baseband part and RF part. The digital baseband part is for A-D and D-A converter. The RF part is for modulation and demodulation. The interface part is used the IEEE 1394 which can support real time transmission of HD video. Fig. 3 shows a top view of the XSUWBWDK UWB transceiver model.

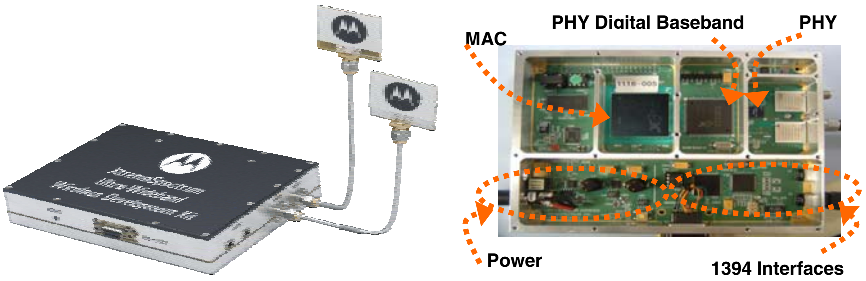


Fig. 3. XSUWBWDK, the development kit of Freescale™

The features of transmitted UWB signal in XSUWBWDK can be represented as

- Average Power Measured on Power Meter: <150 uW or <-8dBm
- Peak-to-Peak Voltage into Antenna: <0.3Volts
- PEP(peak envelope power): <0.225mW

$$PEP = \frac{V_{p-p} \times V_{p-p}}{8 \times 50} \tag{3}$$

The 50 in the denominator of (3) denotes the load impedance and $V_{p\text{-}p}$ is measured peak-to-peak voltage when load impedance is 50 ohms

4 Simulation

4.1 Simulation Environment

Fig. 4 shows an example of HD video transmission scenario. And Fig. 5 depicts the corresponding superframe structure. The PNC can transmit two HD video via stream 3 and stream 4 in CTAP in a superframe. The DEV1 receives the video stream from channel 63 and the DEV 2 also receives it from channel 62.

The XSUWBWDK supports a test mode for performance measurement. In this mode, XSUWBWDK calculates frame rate and FER(Frame Error Rate), while transmit a test frame from transmitter to receiver except PC interface. The frame rate is calculated as the number of correctly received frames per second, and the FER is the number of erroneous frames divided by total number of transmitted frames. The transmitter sends test frame with a rate of 320 frames per second in our simulation.

We consider two test environments: one is for modeling of real office and the other is for empty space modeling. In the real office environment, the UWB system can be influenced by several multipath fading and ISI(Inter symbol Interference)s, just like in a real environment. On the contrary, there exist only a few multipath fading in the empty space environment. Also, we test the LOS(Line of Sight) and NLOS(Not LOS) cases in each case.

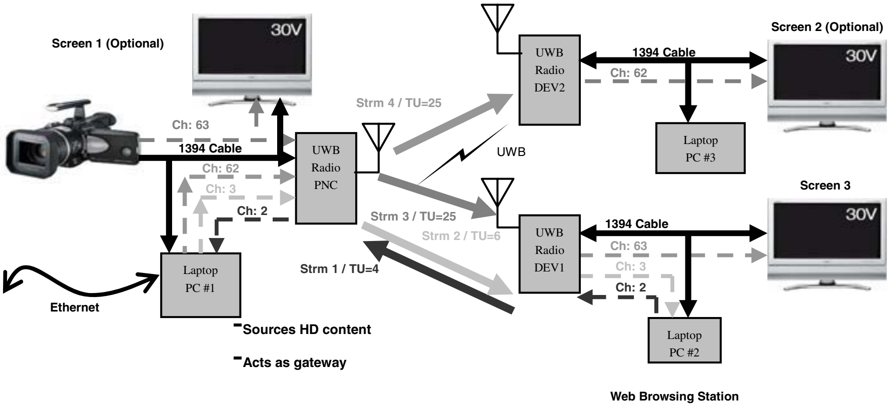


Fig. 4. An example of HD video communication

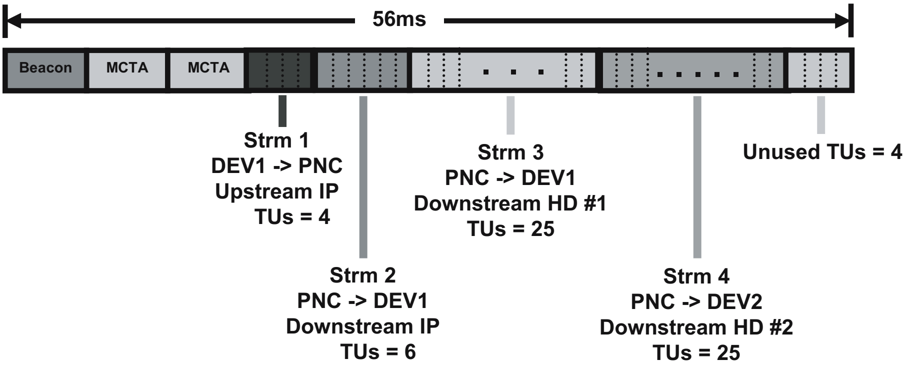


Fig. 5. Transmitted UWB superframe

The position of transmitter and receiver antennas was 80cm from the ground, and the distance from transmitter antenna to receiver antenna was 1m, 3m, 5m, 7m and 9m. In each distance, we measured the frame rate and FER.

4.2 Simulation Results

Fig. 6 and Fig. 7 show the frame rate and FER results. The followings denote the simulation environment cases in our simulation.

- LOS1 : real office environment, line of sight
- NLOS1 : real office environment, not line of sight
- LOS2 : empty space environment, line of sight
- NLOS2 : empty space environment, not line of sight

In case of the real office environment, the HD multimedia frames can be successfully transmitted and received within 7m coverage by using DS-UWB system.

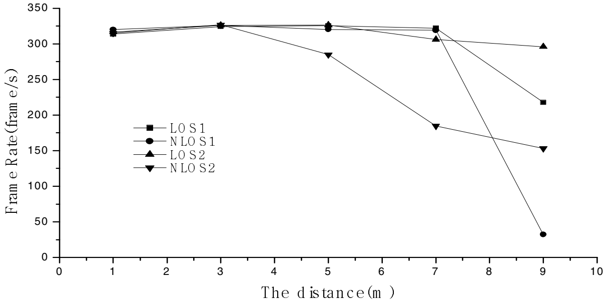


Fig. 6. The frame rates at 1m, 3m, 5m, 7m and 9m

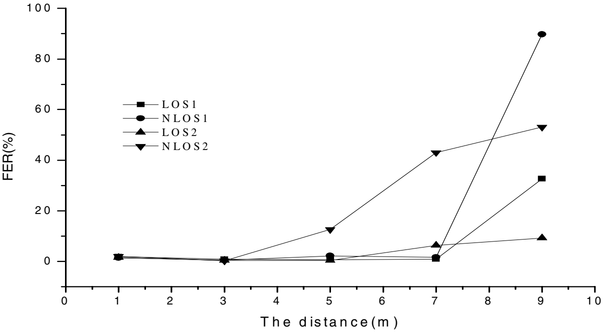


Fig.7. The FERs at 1m, 3m, 5m, 7m and 9m

However, if the distance is farther than 7m, the frame rate was decreased. In case of the empty space environment, the decrease of frame rate was observed from 3m distance. Meanwhile, the FER results in Fig. 7 show the reciprocal characteristic to the frame rate.

5 Conclusion

In this paper, we proposed UWB system as wireless technology for HD multimedia transmission in a home network. We examined the features of the UWB system and compared them with those of the WLAN system. The performance of the UWB system was analyzed by using the development kit, XSUWBWDK of Freescale^{TM}. And we showed that the DS-UWB system can be successfully used for HD multimedia transmission within about 7m coverage. We consider that the UWB system is the most promising candidate for high quality video transmission in a home network since it can afford to provide high data rate and adopt the piconet structure. It is possible to apply the UWB system where the high quality video needs to be shared in wireless environment, for example, there exist one set-top box or DVD player and many display devices in a home. In addition, this system can be used for wireless multimedia communication between computer and beam projector or camcorder. As noted above, the relatively small coverage of UWB system can be overcome via backbone network or repeater.

References

1. Jeyhan Karaoguz, "High-rate wireless personal area network," IEEE communications Magazine, vol. 39, issue 12, pp. 96-102, Dec. 2001.
2. FCC(Federal Communication Commission), 02-48 UWB Report & Order Released 22, Apr. 2002.
3. http://www.ieee802.org/15/pub/TG3a.html
4. Texas Instruments et al., Multi-band OFDM Physical Layer Proposal for IEEE 802.15 Task Group 3a, IEEE P802.15-03/268r2, Nov. 2003.
5. IEEE P802.15 Working Group for Wireless Personal Area Networks(WPANs), DS-UWB Physical Layer Submission to 802.15 Task Group 3a, July 2004.
6. IEEE Std 802.15.3, Wireless Medium Access Control(MAC) and Physical Layer(PHY) Specifications for High Rate Wireless Personal Area Networks(WPANs), June 2003.
7. Wireless Developer Kit Hardware User's Guide, XSUWBDKUG Rev. 1.6 Freescale^{TM} Semiconductor, Jan. 2005.

Extraction of Implicit Context Information in Ubiquitous Computing Environments*

Juryon Paik, Hee Yong Youn, and Ung Mo Kim

Department of Computer Engineering, Sungkyunkwan University,
300 Chunchun-dong, Jangan-gu, Suwon,
Gyeonggi-do 440-746, Republic of Korea
quasa277@gmail.com, {youn, umkim}@ece.skku.ac.kr

Abstract. The evolution of low-cost, networked sensors, often directly internet-enabled, is bringing truly ubiquitous smart environments into daily life. The more ubiquitous middleware platform is intelligent, the greater context information flood problem has been caused. Hence, there have been increasing demands for efficient methods of discovering desirable knowledge from a large collection of context data. But unfortunately, current ubiquitous middleware platforms do not employ appropriate data mining techniques to meet such growing demands. Therefore, this paper aims to propose a new design of ubiquitous middleware platform that enhances context awareness in evolving pervasive environments. We achieve this goal first by incorporating a mining module into our previously suggested middleware platform CALM (Component-based Autonomic Layered Middleware) and then by instantiating the module with an efficient mining algorithm.

1 Introduction

An important requirement of a middleware system to support ubiquitous computing applications is the provision of a highly configurable and adaptive execution environment that dynamically reacts to changes in operating context. Such context-awareness computing work has been carried out by many researchers [11, 2, 5, 3, 4]. Most of them have been worked on defining context-awareness and some of them are mainly focusing on building context-aware applications. However, little has been done in building a framework which supports context data mining leading to useful and accurate information extraction.

Data mining, defined broadly as extracting valuable information and insights from data, may be the untold half of the ubiquitous applications. Given the potentially huge amount of data streamed by live sensors, algorithms to fuse, interpret, augment, and present information will become an increasingly important part of the pervasive environments. Because this data-rich environment does

* This work was supported in part by the University IT Research Center Project (ITRC) and by the Ubiquitous Autonomic Computing and Network Project, 21st Century Frontier R&D Program, funded by the Korean Ministry of Information and Communication.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 456–465, 2006.

not necessarily guarantee an information-rich environment, the efficient mining techniques are required to build truly useful information systems which can serve as the eyes and ears of decision-making process.

With the increasing volume of Extensible Markup Language (XML) data over online environments, the discovery of useful information from a collection of XML documents is currently one of the main research areas occupying the data mining community. Due in large part to its remarkably free-form, XML indeed is rapidly emerging as the current standard incarnation for data representation and exchange. We therefore assume that context factors obtained from various sensors are encoded using XML. Focusing on XML-encoded context data, this paper proposes to improve a previously suggested middleware platform, CALM, to make it more accurately aware of context information. Our main contribution is the first middleware platform equipped with a mining module to improve context awareness. Towards this goal, we incorporate the mining module into the CALM platform and instantiate the mining module with an efficient XML mining algorithm.

The rest of this paper is organized as follows. We begin by reviewing some related research areas in Section 2. We continue in Section 3 with the description of our XML-encoded context data mining scheme embedded into the CALM. Finally, in Section 4 we sum up the main contributions made in our paper and discuss some of our future works.

2 Backgrounds

We first give an overview of the middleware framework CALM for ubiquitous computing environments, and then briefly describe the definitions of context. Because most ubiquitous middlewares manage context information to provide context-aware services, it is important firstly to understand what context is. Thereafter, we describe an XML data model to represent context factors, and review some important definitions related to XML mining.

2.1 CALM: Component-Based Autonomic Layered Middleware

Open systems solutions and techniques have become *de facto* standard for achieving interoperability between disparate, large-scale, legacy software systems. A key technology of open systems solutions and techniques is middleware. Middleware, in general, is used to isolate applications from dependencies introduced by hardware, operating systems, and other low-level aspects of system architecture [6]. Middleware technology has played an important role in facilitating the development of distributed applications by abstracting the network to create a single-system image, and thereby providing network transparency.

While typical middleware platforms clearly benefit the traditional applications in terms of ease of development and robustness, this may not be true in developing ubiquitous applications. Middleware targeted for more traditional applications is not suitable for providing ubiquitous applications with many new properties of ubiquitous computing such as context-awareness and dynamic configuration. Due to the diffusion of ubiquitous computing environments, ubiquitous applications must be agile and react quickly to adapt themselves to highly dynamic environments at runtime.

An important requirement of a middleware platform to support ubiquitous applications is the provision of an adaptive execution environment that dynamically reacts to changes in operating context [3]. By providing adaptation at the middleware layer, the application is relieved from the need to monitor the environment and can leave the adaptation completely to the decision of the middleware. To respond accurately and rapidly to changes in the operating context, the middleware platform should be able to distinguish important context information, such as user preferences, environmental context factors, and so forth, from useless and disposable context factors. This ability of the middleware platform improves context awareness without interfering with habitual work practices.

There are several works to develop ubiquitous middleware infrastructures [3, 6]. In [12] we introduced a C*omponent-based* A*utonomic* L*ayered* M*iddleware* (CALM) system. The CALM is designed to support context-awareness by providing sufficient flexibility to enable active service deployment and reconfiguration in response of rapidly changing contexts. It is composed of a couple of layers and various tools: two internal layers, one external layer, and tools for agent-based applications. Especially, the purposes of two internal layers are as follow: 1) to provide various distributed services based on context-awareness and situation-awareness, 2) to maximize efficiencies of services provided, 3) to let the middleware platform adapt easily to constantly changing context factors, and 4) to take full advantage of diverse agent based services. Thus, the most fundamental processes of the CALM usually occur in the internal layers. Due to lack of space, we will abstain from going into a discussion of the CALM. We refer the interested readers to the paper [12], for a detailed discussion on the platform.

2.2 Definitions on Context

The term "context-aware computing" is commonly understood by those working in ubiquitous/pervasive computing, where it is felt that context is the key in their efforts to disperse and enmesh computing into our lives.

Dey and Abowd [5] defined context as a piece of information that can be used to characterize the situation of a participant in an interaction. Similarly in [2], it is defined as location, environment, time, and identity of people. By sensing context factors, context enabled applications can present context information to users, or modify their behavior according to changes in the environment [9]. Schilit et al. [11] defined three categories of context, which are computing context, user context, and physical context. They emphasized the importance of applications which get adapt themselves to context. However, in real world and live datasets, the context factors change rapidly, and therefore tend to become subjective and very domain specific. The definition of the context given by

Vajirkar et al. [10] is the information regarding objects which supports the entire process from user query to mining. They defined four types of context factors according to the subjectivity of specifications: user, application, data, and data mining.

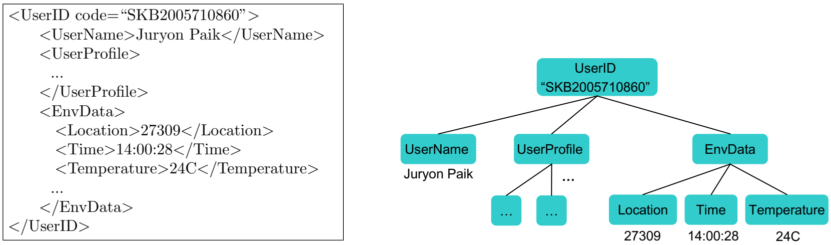
Lack of context-awareness leads to missing a lot of critical and useful information that would affect data mining results. Also, the absence of data mining approach leads to not being able to uncover many hidden factors that would influence the entire system.

2.3 Context Data Representation in XML

In recent years the newly developed XML has gained a tremendous surge of interest from all of the government, industrial and academic sectors. Due to its free-form, XML is rapidly emerging as the current standard incarnation for data representation and exchange of online information. XML represents data as trees, and makes no requirements that the trees be balanced. The only requirements is that the root is the unique node denoting the whole document, the other internal nodes are labeled by tags, and the leaves are labeled by the contents of the document or by the attributes of tags. Thus, XML tree is often called *labeled tree* with a single root.

The facility of free-form permits the generation of composite documents holding information from multiple sources. We obtains context factors from various and distributed sensors for the CALM. Since each sensor has its own format, it is not a trivial work to express, exchange, and store the factors. It is required not only to integrate the factors but also to represent the composite data. Due to the characteristics of remarkably free from, XML is the most adjustable format to describe the context data.

Example 1. The Fig. 1 depicts the XML representation of simple context information of a user. It describes the user's profile, location, and some environmental factors.



```
<UserID code="SKB2005710860">
     <UserName>Juryon Paik</UserName>
     <UserProfile>
       ...
     </UserProfile>
     <EnvData>
        <Location>27309</Location>
        <Time>14:00:28</Time>
        <Temperature>24C</Temperature>
       ...
     </EnvData>
</UserID>
```

(a) Simple context information encoded with XML

(b) XML tree representation

Fig. 1. XML representation of simple context information

2.4 XML Mining

With the continuous growth in XML data sources in the online environment, the data mining community has been motivated to discover useful information from a collection of XML documents. The ability to extract knowledge from them becomes increasingly important.

Definition 1 (Subtree). *Let $T = (N, E)$ be a labeled tree where N is a set of labeled nodes and E is a set of edges. We say that a tree $S = (N_S, E_S)$ is a* ***subtree*** *of T, denoted as $S \preceq T$, iff $N_S \subseteq N$ and for all edges $(u, v) \in E_S$, u is an ancestor of v in T.*

One of the most popular approaches for XML mining is to find subtrees frequently occurring in a set of XML trees.
Let $D = \{T_1, T_2, \ldots, T_i\}$ be a set of trees and $|D|$ be the number of trees in D.

Definition 2 (Support). *Given a set of trees D and a tree S, the frequency of S with respect to D, $freq_D(S)$, is defined as $\Sigma_{T_i \in D} freq_{T_i}(S)$ where $freq_{T_i}(S)$ is 1 if S is a subtree of T_i and 0 otherwise. The* ***support*** *of S w.r.t D, $sup_D(S)$, is the fraction of the trees in D that have S as a subtree. That is, $sup_D(S) = \frac{freq_D(S)}{|D|}$.*

A subtree is called *frequent* if its support is greater than or equal to a minimum value of support specified by a user. This user specified minimum value of support is often called the *minimum support* (*minsup*).

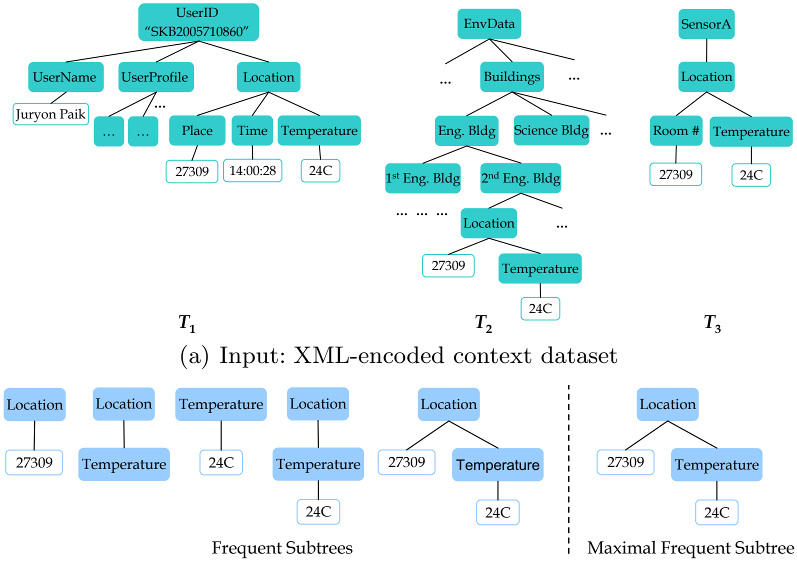
The number of all frequent subtrees can grow exponentially with an increasing number of trees in D, and therefore mining all frequent subtrees becomes infeasible for a large number of trees.

Definition 3 (Maximal Frequent Subtree). *Given some minimum support σ, a subtree S is called* ***maximal frequent*** *w.r.t D iff:*

i) the support of S is not less than σ, i.e., $sup_D(S) \geq \sigma$.
ii) there exists no any other σ-frequent subtree S' w.r.t. D such that S is a subtree of S'.

The definitions given above are commonly used to discover useful information from a collection XML trees. For more details of the XML mining and the extended discussion, the readers may refer to [13, 1, 8].

Example 2. An example of a set of labeled trees D with various context factors is shown in Fig. 2(a). At a glance three context factors obtained from sensors are different from each other and it seems that there is no similarity among them. However, when a minimum support value is between 0.6 and 1, the important hidden information is discovered, as illustrated in Fig. 2(b). With a sufficient reliability more than 60%, we can get to know the commonly-occurring context information; the preferred temperature of location 27309 is apparently 24C. Also with the same reliability, we find the implicit relations between context factors; the Location factor is obtained always together with the Temperature factor.



(a) Input: XML-encoded context dataset

(b) Output: frequent subtrees and maximal frequent subtree

Fig. 2. Finding maximal frequent subtrees from a context dataset, when minimum support is given $0.6 < \sigma \leq 1$

3 CALM Platform Equipped with XML Mining Module

Middleware frameworks for ubiquitous computing obtain vast streams of context factors from various sensors. However, the emergence of novel technologies with ability to generate large amounts of data has not been matched with the ability to represent and exploit the generated data. Due to this mismatch, the decision making required for successful context-awareness at middlewares is actually increasingly difficult. Even worse, implicit context factors are normally ignored because decision making process is mostly performed on the explicit context factors captured via sensors.

In this section, we propose the middleware platform CALM by incorporating a mining module into it and then by instantiating the module with an efficient mining algorithm EXiT-B. To the best of our knowledge, none of the existing ubiquitous middleware platforms has addressed the actual usage of mining algorithms.

3.1 XML-Encoded Context Data Mining Module

As mentioned in earlier section, we adopt an XML data model to store and represent context factors. There is some existing middleware framework [4] which stores its context factors in XML format. However, a significant difference exists between their representation and ours. The former predefines a schema of XML data, which means when context factors are stored, they must follow predefined types, number of tags, and etc. It gives some restrictions on describing the highly various context factors. Meanwhile, the latter has no formal regulatory structure. This implies that any types of XML data can be used, except only the invalid ones. The more XML data have free structures, the more it is difficult to find hidden information. Due to the trade-off, the XML data having no schemas is more required an efficient XML mining algorithm than the data having schemas is. This is the reason why we have to improve the learning and inference module in CALM by embedding a some special module.

Fig. 3 shows the improved learning module for CALM. The left side of the figure depicts a simplified CALM structure only focusing on purely functional parts for context-awareness. From the figure we can see that context factors are monitored, aggregated, interpreted, and finally both stored into storages and provided for the learning and inference module. The learning and inference module plays an absolutely key role in the middleware by deducing new and relevant information to application(s) and user(s) from various sources of context data. However, the accuracy of learning and inference module can be degraded by flood of useless, temporary, and most of all, hidden context information. Thus, we add an auxiliary albeit important module, called *Context Data Mining module* shown on the right side of Fig. 3 to support the learning and inference module. The newly embedded module performs the XML mining to find implicit but useful context information, and then provides the valuable knowledge for appropriate and accurate decision-making.

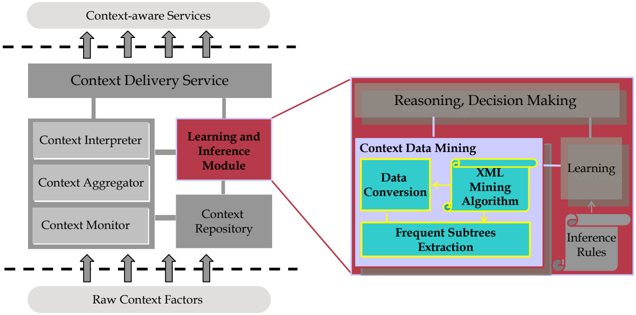


Fig. 3. Embedded context data mining module into learning module of CALM

An efficient and scalable mining algorithm is required in order to manage and extract the information from a huge amount of context factors. Fortunately, in [7, 8], we suggested the algorithm named EXiT-B which enables efficient and scalable extraction of maximal frequent subtrees. Due to the simplified task of finding maximal frequent subtrees, the context data mining module efficiently discovers useful information such as commonly occurring user preferences, interests, or behaviors.

3.2 Algorithm EXiT-B

The purpose of the algorithm EXiT-B is to efficiently find frequent subtrees, especially maximal frequent ones, from a given set of trees. The EXiT-B consists of three functions: *genBitSeq, calFreSet, and maxSubtree.* The *genBitSeq* function represents each tree by a set of bit sequences through assigning an n-bit binary code to each node label and concatenating the codes on the same path. The function *calFreSet* creates and maintains a specially devised data structure called, PairSet, to avoid join operations entirely during the phase of generating maximal frequent subtrees, and reduce the number of candidate subtrees from which frequent subtrees are derived. It uses a collection of bit sequences and minimum support as inputs, and produces frequent n-bit codes to be stored in the special structure PairSets. The *maxSubtree* extracts maximal frequent subtrees incrementally based on the n-bit binary codes stored in the frequent PairSets produced by the function *calFreSet.*

There are two unique features that distinguish EXiT-B to other XML mining algorithms: bit sequences representation of XML trees and PairSet data structure for storing each binary code along with its tree indexes.

Bit Sequences. Let L be a set of labeled nodes in a set of trees D. To represent each tree as a set of bit sequences; firstly, an unique n-bit binary code is randomly assigned to every labeled node. The same n-bit code must be assigned to the labeled nodes with the same name. Let $|L|$ be a total number of labeled nodes in L. Then, the bit size of n is $\lceil \log_2 |L| \rceil$. For instance, we need only 4-bit binary code to represent 16 different node labels. Secondly, it is a need to concatenate all n-bit binary codes on the same path from the root to each leaf node in a tree. We call the concatenated n-bit binary codes for each path *bit sequence.*

PairSet. Another features of EXiT-B is the use of a new hierarchical data structure named *PairSet.* It is a kind of array structures consisting of (*key, tlist*) pairs, where *key* is a unique n-bit code at each depth d of every tree in D and *tlist* is a list containing some tree indexes of which the corresponding key is located at the depth d.

We refer the readers to the papers [7, 8] for a detailed discussion on the algorithm.

3.3 Evaluation of EXiT-B for Context Data

In this subsection, we carried out a couple of experiments to show the time consumption of EXiT-B over context factors. Unfortunately, we could not directly test the proposed module under the CALM platform, because we are currently working on building a prototype for it [12]. Thus, only for this paper we generated the artificial context dataset whose structure is basically XML-based tree structure.

The experiments were done on an AMD Athlon 64 3200+ PC with 1GB main memory, running Windows XP. The mining algorithm was implemented in JAVA. We limit the total number of different context factors to no more than

100, and set both the maximum branch factor of each node and the maximum depth of each XML tree to 3. The experiment was done with various numbers of XML trees for various values of minimum supports. The result of the experiment is shown in Fig. 4. The execution time increases in a quadratic manner with the increase of the number of XML trees. Also the variation of minimum support does not degrade the performance of the algorithm EXiT-B. Since the execution time to discover the valuable but hidden context information is not adversely affected by the number of context factors and the range of minimum support, the experiment at least indicates that the load of managing and processing the mining module for the middleware platform will not cause the significant time penalty no matter how many context data is being used in the platform.

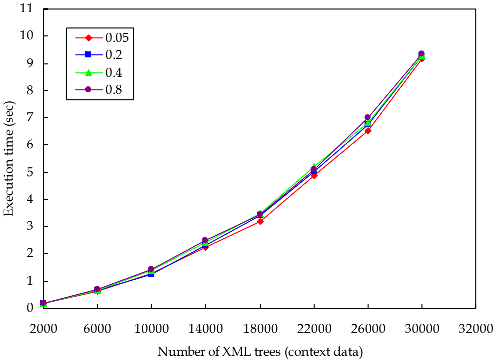


Fig. 4. Time consumption

4 Conclusion

In this paper we introduced the improved CALM platform, which, as far as we know, is the first middleware platform framework being addressed the actual usage of XML mining algorithm EXiT-B in evolving pervasive environments. To this end we developed the auxiliary albeit important module, context data mining module, whose purpose is to find the important but implicit context information for accurate decision-making of the CALM. For more reliable and accurate tests of the proposed scheme, we are currently working on building a prototype of the CALM.

References

1. Asai, T., Abe, K., Kawasoe, S., Arimura, H., Sakamoto, H., Arikawa, S.: Efficient substructure discovery from large semi-structured data. Proceedings of the 2nd SIAM International Conference on Data Mining (2002) 158–174
2. Brown, P. J.: The Stick-e Document: a Framework for Creating Context-Aware Applications. Electronic Publishing (1996) 259–272

3. Chan, Alvin T. S., Chuang, S-N.: MobiPADS: A Reflective Middleware for Context-Aware Mobile Computing. IEEE Transactions on Software Engineering 29(12) (2003) 1072–1085
4. Choi, J., Shin, D., Shin, D.: Research and Implementation of the Context-Aware Middleware for Controlling Home Appliances. IEEE Transactions on Consumer Electronics 51(1) (2005) 301–306
5. Dey, A. K., Abowd, G. D.: Towards a better understanding of Context and Context-Awareness. Workshop on the What, Who, Where, When, Why and How of Context-Awareness at Conference on Human Factors in Computing Systems (2000). Lecture Notes in Computer Science, Vol. 1707. Springer-Verlag, Berlin Heidelberg New York (1999) 304–307
6. Lawson, J., Raines, R., Baldwin, R., Hartrum, T., Littlejohn, K.: Modeling Adaptive Middleware and Its Application to Military Tactical Datalinks. IEEE Military Communications Conference (2004) 975–980
7. Paik, J., Shin, D. R., Kim, U. M.: EFoX: a Scalable Method for Extracting Frequent Subtrees. Proceedings of the 5th International Conference on Computational Science. Lecture Notes in Computer Science, Vol. 3516. Springer-Verlag, Berlin Heidelberg New York (2005) 813–817
8. Paik, J., Won, D., Fotouhi, F., Kim, U. M.: EXiT-B: A New Approch for Extracting Maximal Frequent Subtrees from XML Data. Proceedings of the 6th International Conference on Intelligent Data Engineering and Automated Learning. Lecture Notes in Computer Science, Vol. 3578. Springer-Verlag, Berlin Heidelberg New York (2005) 1–8
9. Salber, D., Dey, A. K., Orr, R. J., Abowd, G. D. Designing for Ubiquitous Computing: A Case Study in Context Sensing. GVU Technical Report GIT-GVU 99-29. http://smartech.gatech.edu:8282/dspace/handle/1853/3396
10. Vajirkar, P., Singh, S., Lee, Y.: Context-Aware Data Mining Framework for Wireless Medical Application. Proceedings of the 14th International Conference on Database and Expert Systems Applications. Lecture Notes in Computer Science, Vol. 2736. Springer-Verlag, Berlin Heidelberg New York (2003) 381–391
11. Shilit, B., Adams, N., Want, R.: Context-Aware computing applications. Proceedings of IEEE Workshop on Mobile Computing Systems and Applications (1994) 85–90
12. You, Y. K., Han, S., Song, S. K., Youn, H. Y.: CALM: An Intelligent Agent-based Middleware Architecture for Community Computing. The 4th Workshop on Adaptive and Reflective Middleware (2005)
13. Zaki, M. J.: Efficiently mining frequent trees in a forest. Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data mining (2002) 71–80

Convergence of Context-Awareness and Augmented Reality for Ubiquitous Services and Immersive Interactions

Jae Yeol Lee[1,*], Gue Won Rhee[1], Hyun Kim[2], Kang-Woo Lee[2], Young-Ho Suh[2], and Kwangsoo Kim[3]

[1] Department of Industrial Engineering, Chonnam National University, South Korea
[2] Software Robot Research Team, ETRI, South Korea
[3] Department of Industrial and Management Engineering, POSTECH, South Korea
jaeyeol@chonnam.ac.kr

Abstract. Computing paradigm is moving toward context-aware and ubiquitous computing in which devices, software agents, and services are all expected to seamlessly integrate and cooperate in support of human objectives. Augmented reality (AR) can naturally complement ubiquitous computing by providing an intuitive and collaborative interface to a three-dimensional information space embedded within physical reality. This paper presents a framework and its applications for the convergence of context-awareness and augmented reality, which can support a rich set of ubiquitous services and immersive interactions. The framework provides a common data model for different types of context information from external sensors, applications and users. It also offers the software framework to acquire, interpret and disseminate context information. Further, it utilizes augmented reality for providing immersive interactions by embedding virtual models onto physical models, which realizes bi-augmentation between physical and virtual spaces.

1 Introduction

Context-aware and ubiquitous computing is a vision of our future computing lifestyle in which computer systems seamlessly integrate into our everyday lives, providing services and information in anywhere and anytime fashion [1,11]. Context-aware and ubiquitous systems are computer systems that can provide relevant services and information to users by exploiting contexts. By contexts, we mean information about locations, software agents, users, devices, and their relationships [5,13].

Augmented reality (AR) can naturally complement ubiquitous computing by providing an intuitive and collaborative interface to a three-dimensional information space embedded within physical reality [2,10]. Correspondingly, the human-computer interfaces and interaction metaphors originating from AR research have proven advantageous in a variety of real-world ubiquitous application scenarios, such as industrial assembly and maintenance, location-based intelligent systems, navigation aides, and computer-supported cooperative work [2,6,12].

[*] Corresponding author: 300 Yongbong-dong, Buk-gu, Gwangju, 500-757, South Korea.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 466 – 474, 2006.

Recently, the concept of service robots comes to us as if they were digital appliances and information services [4]. The Korean government has strategically promoted the development of new concept of the network-based intelligent robot, which is called as URC (Ubiquitous Robotic or Reality Companion) [8]. To realize true URC, it is recognized that the software infrastructure plays an important role to expand the robot functions and services, improve the context-awareness in the external environment, and enhance the robot intelligence.

Although context-aware and ubiquitous computing is very popular in the areas of building intelligent meeting rooms, supporting intelligent robots, providing smart spaces for easy living [3,5,9], a more sophisticated research is still needed, which should combine context-aware computing with more natural and intuitive interfaces like augmented reality for providing human-centered services and immersive interactions. Note that the need for such requirements is increasing rapidly so that a neutral framework or middleware must be provided to support context management and execution for various ubiquitous applications.

This paper presents a framework and its applications for the convergence of context-awareness and augmented reality, which can support ubiquitous services and immersive interactions. The framework provides a common data model for different types of context information from external sensors, applications and users in the environment. It also offers the software framework to acquire, interpret and disseminate context information. Further, it utilizes augmented reality for providing more relevant and immersive interactions and collaborations by embedding virtual models onto physical models considering contexts. Moreover, human interactions based on AR not only feedback to existing contexts, but also generate new contexts, which can realize bi-augmentation between physical and virtual spaces. It has been applied to various applications such as intelligent home and simulation, software robot, and ubiquitous maintenance service. The remainder of the paper is organized as follows. Section 2 overviews the proposed system. Section 3 presents how to maintain contexts and apply them to augmented reality in ubiquitous environments. Section 4 shows some implementation results. Finally, Section 5 concludes with some remarks.

2 System Overview

The primary objective of this research is to propose a generic framework that supports the convergence of context-awareness and augmented reality for ubiquitous services and immersive interactions as shown in Fig. 1. The framework has been built on the three layers: 1) U-interface layer, 2) U-context layer, and 3) AR interaction layer. The U-context layer maintains contexts from various resources such as devices, people, environment, etc. Further, the U-context broker facilitates reasoning and execution of those contexts. The U-context layer is based on CAMUS (Context-Aware Middleware for URC System) which is a middleware for supporting the context-awareness of ubiquitous services such as devices, sensors, and sobots (software robots). The U-interface layer supports bi-interactions between physical devices (or software components) and the U-context layer. Thus, all the devices can be easily registered, searched, and executed by the CAMUS-enabled broker. The AR interaction layer provides more realistic and human-oriented services using an AR technique. It is linked to the U-interface and

U-context layers for context acquisition and reasoning, and graphical information gathering and synchronization. Thus, the three layered framework can support various kinds of ubiquitous services and interactions such as context-aware adaptation to the environment and human-centered AR-enabled interactions and simulations.

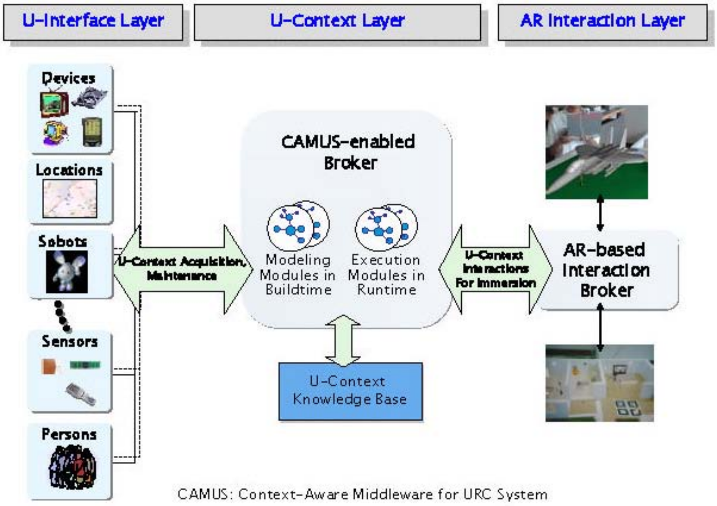

U-Interface Layer
U-Context Layer
AR Interaction Layer
Devices
Locations
Sobots
Sensors
Persons
U-Context Acquisition, Maintenance
CAMUS-enabled Broker
Modeling Modules in Buildtime
Execution Modules in Runtime
U-Context Interactions For Immersion
AR-based Interaction Broker
U-Context Knowledge Base
CAMUS: Context-Aware Middleware for URC System

Fig. 1. CAMUS-enabled ubiquitous service framework

3 Convergence of Context-Awareness and Augmented Reality

This section explains how contexts are managed and reasoned to provide more relevant and ubiquitous services. It also discusses how to utilize augmented reality for executing context-aware interactions.

3.1 CAMUS-Based Context Management and Execution

The CAMUS-based middleware consists of two parts as shown in Fig. 2: 1) build-time components for ubiquitous space modeling and 2) runtime components for task execution [8]. The build-time components are used for registering and managing physical sensors, ubiquitous services, environments, users, and tasks. For example, the sensor modeler offers means for mapping sensors of the physical space into sensor services of the cyber space, extracting context information from the sensors and supplying the information to the task engine. The task modeler supports the modeling of context information specific to a task and the description of the rules necessary to perform the task. Then, the built tasks are executed by the runtime components. Among the runtime components, the task manager plays the main role in executing tasks. It initiates individual tasks and manages on-going task processes. The task engine executes the actual tasks considering the situation. It has an inference engine to process facts and rules supplied by a task. In CAMUS, we applied JESS as an inference engine [7].

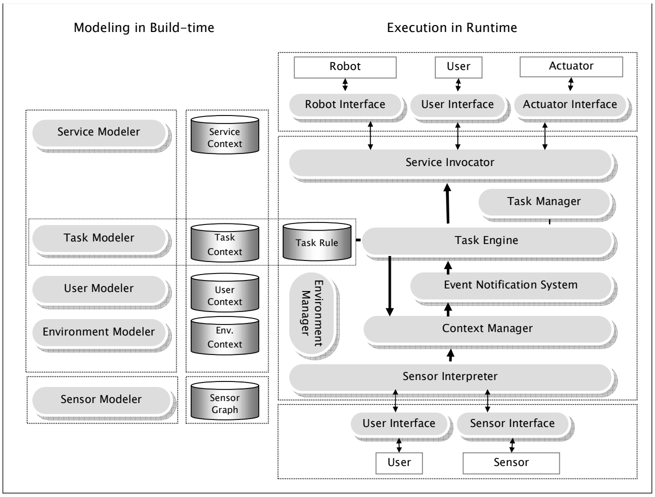


Fig. 2. Build-time and runtime components of CAMUS

The context manager manages context information which is represented by the *Universal Data Model* (UDM). When context information in the environment is changed, the context manager propagates the events to the event notification system. Finally, events are delivered to the task engine to supply the necessary context information required for the task execution. The task engine executes the actual tasks considering the situation. UDM represents context information as nodes and associations between them. The node represents an entity such as person, place, task, service, etc. Every node has its unique ID and type. There are special nodes which have "valued" type (grey circle in Fig. 3). Because the association starts and stops at a node, the association has direction, defined by which node is a from-node (Fnode) and which node is a to-node (Tnode).

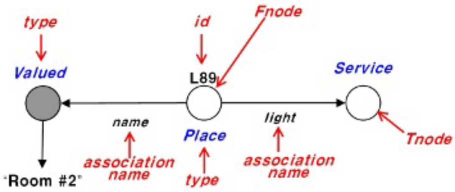


Fig. 3. An example of UDM

Each task is described by using PLUE (Programming Language for Ubiquitous Environment). It is basically an extension of Java programming language, and in fact, its compiler is a pre-processor of the Java compiler. Fig. 4 shows an example of the PLUE program that implements a smart-room application. It is generally agreed that 'rule-based programming' plays a key role in presenting proactive, intelligent, and invisible services to service requestors. Therefore, it is a main goal of the PLUE design to introduce the rule-based programming feature into a conventional procedural programming language. Rules in PLUE can be augmented with an event expression so that they are fired only when the expected events are received. That is, they are Event-Condition-Action (ECA) rules and intensively used in the domain where applications are needed to react to environmental changes. As shown in the above example, the expression 'on ($place.temperature::ValueChanged e)' describes the event that fires this rule and the 'if (count($place.resident) > ......' describes the condition of the rule, and finally the rest of the rule shows action part. The rule in the example is read as "whenever the temperature of a room gets higher than the allowed, turn on the air conditioner of the room and set its temperature to the average of the residents' preferences." The conventional Java method calls are allowed in rule expression; 'getDesiredTemperature()' and 'getHighestAllowed()" are used in this example.

```
Task SmartRoom {
   //
   // When this room becomes hot, turn on the air conditioner in the room, and set its
   // temperature to the average of residents' preferences.

   on ($place.temperature::ValueChanged e) {
      if (count($place.resident) > 0 && e.value < $task.highestAllowed) {
        if (!$place.air_conditioner.power) {
          $place.airconditioner.power = true;
        }
        $place.airconditioner.desired_temp = $task.desiredTemperature;
      }
   }

   // calculates the average of residents' preferences
   public int getDesiredTemperature() {
      double avgHigh = avg($place.resident.prefered_temp.highest);
      double avgLow = avg($place.resident.preferred_temp.lowest);
      return (int)(avgHigh - avgLow)/2;
   }

   // calculate the average of residents' highest preferences
   public int getHighestAllowed() {
      return (int)avg($place.resident.preferred_temp.highest);
   }
}
```

Fig. 4. An example of the PLUE program

Moreover, the hierarchical representation of contexts among tasks, persons, and devices is also maintained. It consists of five layers as shown in Fig. 5. The bottom layer includes a range of mobile and fixed devices [1]. The second layer contains device proxies, which every device has. The third layer is the user-proxy layer. Every user has a personal user proxy. This layer can store applications and a user's state. The fourth layer is the task layer where each task is shared. The fifth layer is related to UDM of the registered proxies. This representation can make ubiquitous services be more adoptable to a dynamic changing environment.

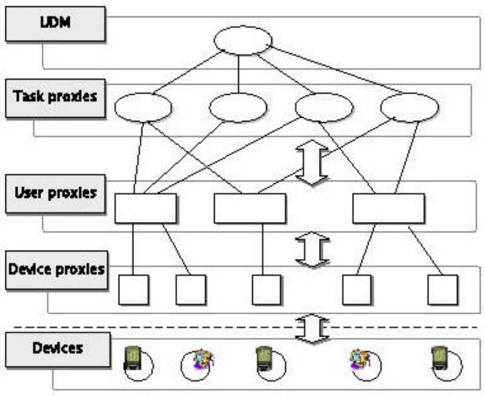


Fig. 5. Hierarchical representation of context relations among tasks, users, and devices

3.2 Augmented Reality-Based Immersive Interactions

The AR-based interaction broker consists of 4 major modules as shown in Fig. 6: U-context binding module, U-interface binding module, tracking module, and rendering module. Internally, the tracking module and rendering module support AR applications. The tracking module is based on a marker-based tracking technique, also supporting multi-marker tracking capabilities. In this research, ARToolkit has been utilized [2,10,12].

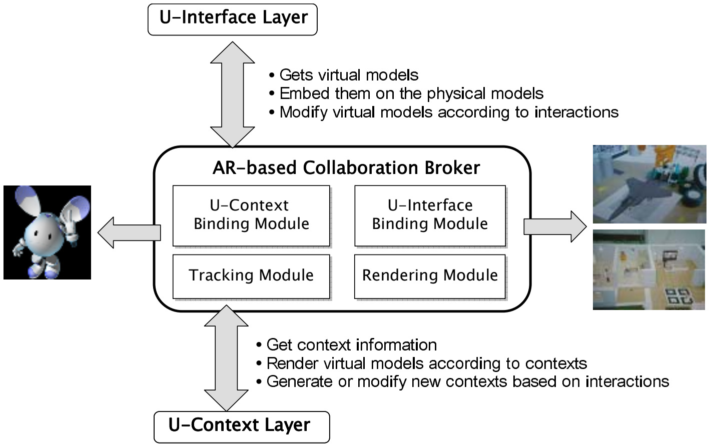
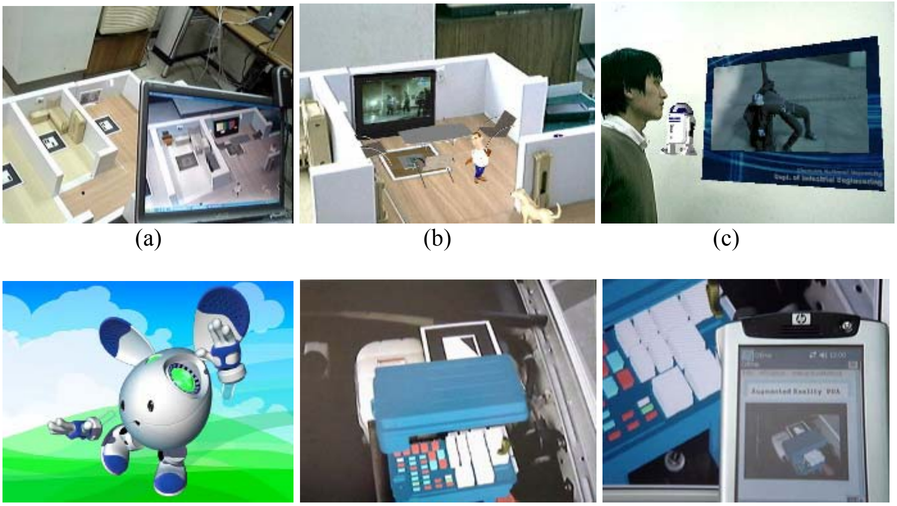

U-Interface Layer
• Gets virtual models
• Embed them on the physical models
• Modify virtual models according to interactions
AR-based Collaboration Broker
U-Context Binding Module
U-Interface Binding Module
Tracking Module
Rendering Module
• Get context information
• Render virtual models according to contexts
• Generate or modify new contexts based on interactions
U-Context Layer

Fig. 6. Modules of the AR interaction layer

The rendering module embeds the 3D virtual reality of service and context information onto the physical reality image synchronized by the tracking module. Externally, the U-Context binding module and U-Interface binding module are used to communicate with the U-Context layer and U-Interface layer for context and service information retrieval and synchronization. The U-Interface binding module receives virtual models from the U-Interface broker, then, applies various interactions, and finally feedbacks the interactions to the U-Interface broker, which can modify the original model or generate new models. Similarly, the U-Context binding module gets context information from the U-Context layer and then embeds the contexts to AR. Further, it also feedbacks new contexts generated from AR interactions to the U-Context layer. Moreover, the U-Context broker queries and reasons about contexts, and sends the derived contexts to the U-Context interface module, which again applies them to AR.

4 System Implementation

This section explains how the proposed framework can be integrated and applied to various ubiquitous and context-aware applications. To illustrate the benefits of the proposed approach, we present the following three application results: 1) intelligent home and simulation, 2) sobot assisted ubiquitous multi-media services, and 3) ubiquitous car services by augmenting virtual prototypes into real cars as shown in Fig. 7.



(a) (b) (c)

(d) (e) (f)

Fig. 7. Implementation of the convergence of context-awareness and augmented reality

Fig. 7(a) & (b) show how the AR-based technique can be applied to simulate an intelligent home. To verify the effectiveness of utilizing AR, we constructed a miniaturized intelligent home. However, we concluded that it would be quite difficult to model and simulate dynamic objects and devices, which limits realistic context-aware experiments. On the other hand, using the AR technique, we realized that AR could be effectively used by dynamically embedding virtual models into the physical environment, which can simulate real environments, although all kinds of ubiquitous devices are not equipped with. Fig. 7(c) & (d) show how a sobot can be assisted to provide AR-based multi-media services and interactions. With the help of the sobot, a user can activate an AR-enabled movie player using a RFID tag which generates an event to CAMUS and then CAMUS executes a task for playing a movie. In addition, the user can communicate with the sobot for various kinds of ubiquitous services such as weather forecasting, games, and news services. Fig. 7(e) & (f) show how the framework can be applied to supporting ubiquitous car services by augmenting virtual prototypes into real cars. By utilizing context awareness, users can have various kinds of car maintenance services in ubiquitous environments regardless of their devices and situations. Considering the implemented results, we realized that ubiquitous environments can be much more realistic, interactive, and immersive if the AR technique can be fully utilized.

5 Conclusion

The convergence of context-awareness and augmented reality has been proposed for supporting various ubiquitous applications such as intelligent home and its simulation, sobot-assisted immersive environment for multi-media services and secretary, and ubiquitous car maintenance services. The framework provides a common data model for different types of context information from external sensors, applications and users in the environment. It also offers the software framework to acquire, interpret and disseminate context information. Further, it utilizes augmented reality for providing more relevant and immersive interactions and collaborations by embedding virtual models onto physical models considering contexts, which can realize bi-augmentation between physical and virtual spaces. In conclusion, the convergence can be very effectively utilized for: 1) seamless interaction between real and virtual environments, 2) providing context-awareness, 3) presenting spatial cues for various kinds of interactions such as product development and intelligent home, and 4) providing the ability to transit smoothly between reality and virtuality.

Acknowledgements

This research was supported by the Korean Ministry of Information and Communication (Grant No. A1100-0502-0028) and the Korea Science & Engineering Foundation (Grant No. R01-2003-000-10171-0).

References

1. Anhalt, J., Smailagic, A., Siewiorek, D.P., Gemperle, F., Salber, D., Weber, S., Beck, J., Jennings, J.: Toward context-aware computing: experiences and lessons. IEEE Intelligent Systems 16 (2001) 38-46.
2. Billinghurst, M. Kato, H.: Collaborative augmented reality. Communications of the ACM 45(2002) 64-70
3. Brumitt, B., Meyers, B., Krumm, J., Kern, A., Shafer, S.: EasyLiving: technologies for intelligent environments. HUC2000 LNCS 1927, Springer-Verlag (2000) 12-29

4. Bruyninckx, H.: Open robot control software: the OROCOS project. IEEE International Conf. on Robotics and Automation (2001) 2523-2528
5. Chen, H., Finin, T., Joshi, A., Kagal, L., Perich, F., Chakraborty, D.: Intelligent agents meet the semantic web in smart spaces. IEEE Internet Computing 8(2004) 69-79
6. Doil, F., Schreiber, W., Alt, T., Patron, C.: Augmented reality for manufacturing planning. Proc. of the workshop on Virtual Environments 2003 (2003) 71-76
7. Freeman-Hill, E.: Jess Manual, Sandia National Laboratories, Livermore, CA, USA, (1997)
8. Kim, H., Cho, Y.-J., Oh, S.-R.: A middleware supporting context-aware services for network-based robots. IEEE Workshop on Advanced Robotic and its Social Impacts, Nagoya, Japan (2005)
9. Kindberg, T., et al.: People, places, things: web presence for the real world, Mobile Networks and Applications. Kluwer Academic Publishers (2002) 365-376
10. Lee, J.Y., Seo, D.W.: A context-aware and augmented reality-supported service framework in ubiquitous environments. EUC2005 LNCS 3823 (2005) 258-267
11. Suzuki, G. *et al.*: u-Photo: Interacting with pervasive services using digital still images. Pervasive 2005 LNCS 3468 (2005) 190-207
12. Wagner, D., Pintaric, T., Ledermann, F., Schmalstieg, D.: Towards massively multi-user augmented reality on handheld devices. Pervasive 2005 LNCS 3468 (2005) 208-219
13. Wang, X., Dong, J.S., Chin, C.Y., Semantic space: an infrastructure for smart spaces. IEEE Pervasive Computing 3(2004) 32-39

An Adaptive Fault Tolerance System for Ubiquitous Computing Environments: AFTS

Eung Nam Ko

Department of Information & Communication, Baekseok University,
115, Anseo-Dong, Cheonan, ChungNam, 330-704, Korea
ssken@bu.ac.kr

Abstract. This paper presents the design of the AFTS(An Adaptive Fault Tolerance System), which is running on situation-aware middleware. Situation-aware middleware provides standardized communication protocols to interoperate an application with others under dynamically changing situations. Since the application needs of middleware services and computing environment (resources) keep changing as the application change, it is difficult to analyze: whether it is possible that all Quality of Service (QoS) requirements are met, and what QoS requirements have tradeoff relationships. In this paper, we propose a QoS resource error detection-recovery model called "AFTS" for situation-aware middleware. An adaptive Video On Demand (VOD) system is used as an illustrative example of the AFTS model and its resource error detection-recovery.

1 Introduction

In 1991, Mark Weiser, in his vision for the 21st century computing, described that ubiquitous computing, or pervasive computing, is the process of removing the computer out of user awareness and seamlessly integrating it into everyday life. We can describe ubiquitous computing as the combination between mobile computing and intelligent environment is a prerequisite to pervasive computing[1]. Context awareness(or context sensitivity) is an application software system's ability to sense and analyze context from various sources; it lets application software take different actions adaptively in different contexts[2]. In a ubiquitous computing environment, computing anytime, anywhere, any devices, the concept of situation-aware middleware has played very important roles in matching user needs with available computing resources in transparent manner in dynamic environments [3,4]. Although the situation-aware middleware provides powerful analysis of dynamically changing situations in the ubiquitous computing environment by synthesizing multiple contexts and users' actions, which need to be analyzed over a period of time, it is difficult to analyze Quality of Service (QoS) of situation-aware applications because the relationship between changes of situations and resources required to support the desired level of QoS is not clear. Thus, there is a great need for situation-aware middleware to be able to predict whether all QoS requirements of the applications are satisfied and analyze tradeoff relationships among the QoS requirements, if all QoS requirements cannot be satisfied to determine a higher priority of QoS requirements.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 475 – 482, 2006.

Our AFTS model is to present the relationship of missions, actions, QoS and resources. AFTS model is used to detection and recover the QoS resource errors among actions. Our approach has distinct features such as the SoC principle support and dynamism of situation-aware support. For existing QoS management techniques: In OS-level management, QoS management schemes are limited to CPU, memory, disk, network, and so on [5,6]. In application-level management, a limitation of the schemes is in monitoring states of necessary resources from applications steadily or periodically. To extend the limitation, QoS is managed in the middleware level to satisfy integrated QoS of several applications over the network [7,8,9]. However, these approaches are also limited and not flexible in dynamically changing situations, comparing with our situation-aware QoS management using the AFTS model.

2 The Context: Situation-Aware Middleware

A conceptual architecture of situation-aware middleware based on Reconfigurable Context-Sensitive Middleware (RCSM) is proposed in [2]. Ubiquitous applications require use of various contexts to adaptively communicate with each other across multiple network environments, such as mobile ad hoc networks, Internet, and mobile phone networks. However, existing context-aware techniques often become inadequate in these applications where combinations of multiple contexts and users' actions need to be analyzed over a period of time. Situation-awareness in application software is considered as a desirable property to overcome this limitation. In addition to being context-sensitive, situation-aware applications can respond to both current and historical relationships of specific contexts and device-actions. All of RCSM's components are layered inside a device. The Object Request Broker of RCSM (R-ORB) assumes the availability of reliable transport protocols; one R-ORB per device is sufficient. The number of ADaptive object Containers (ADC)s depends on the number of context-sensitive objects in the device. ADCs periodically collect the necessary "raw context data" through the R-ORB, which in turn collects the data from sensors and the operating system. Initially, each ADC registers with the R-ORB to express its needs for contexts and to publish the corresponding context-sensitive interface. RCSM is called reconfigurable because it allows addition or deletion of individual ADCs during runtime (to manage new or existing context-sensitive application objects) without affecting other runtime operations inside RCSM. An example of *SmartClassroom* is illustrated in [2]. However, it did not include QoS support in the architecture. In this paper, we focus on how to represent QoS requirements in situation-aware middleware. In the next subsection, we will present a conceptual model for QoS requirements representation in situation-aware middleware.

3 The AFTS Model

The conceptual architecture of the AFTS model is described in section 3.1, and its model description language is proposed in section 3.2. An adaptive QoS management algorithm is presented in section 3.3.

3.1 Overview of the AFTS Model

Our proposed AFTS model aims at supporting adaptive **Q**oS requirements defined in application-level **M**issions described by a set of **A**ctions of objects by reserving, allocating, and reallocating necessary **R**esources given dynamically changing situations.

A high-level AFTS conceptual architecture to support adaptive QoS requirements is shown in Figure 1. Situation-aware Manager (SM), Resource Manager (RM), and QoS Management Agent (QMA) are the main components shown in Situation-Aware Middleware box in Figure 1. Applications request to execute a set of missions to Situation-aware Middleware with various QoS requirements. A Situation-aware Manager analyzes and synthesizes context information (e.g., location, time, devices, temperature, pressure, etc.) captured by sensors over a period of time, and drives a situation. A Resource Manager simultaneously analyzes resource availability by dividing requested resources from *missions* (i.e., a set of object methods, called *actions*) by available resources. It is also responsible for monitoring, reserving, allocating and deallocating each resource. Given the driven situations, A QoS Management Agent (QMA) controls resources when it met errors through the Resource Manager to guarantee requested QoS requirements.

If there are some error resource due to low resource availability, QMA performs QoS resource error detection-recovery. RM resolves the errors by recovering resources for supporting high priority missions. To effectively identify and resolve QoS conflicts, we need to capture the relationships between mission, actions, its related QoS requirements, and resources. For this reason, we also propose a model description language for AFTS in Section 3.2.

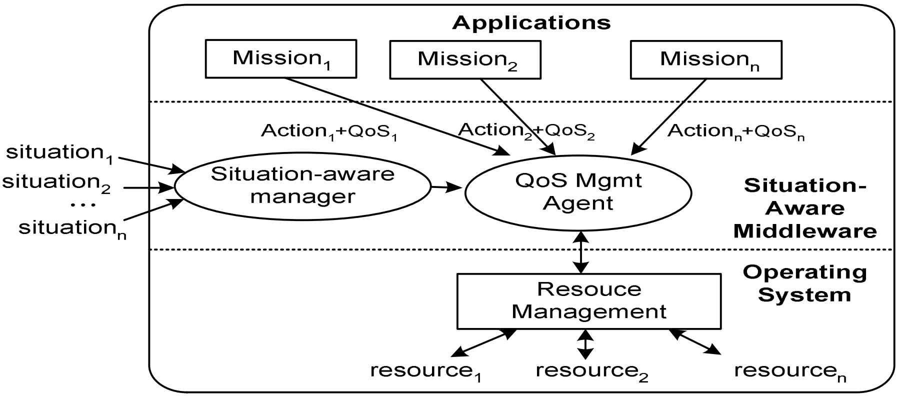


Fig. 1. Overview of Our Proposed AFTS Model

3.2 AFTS Model Description Language

The AFTS model effectively represents the relationships among missions, QoS, actions, and resources for detecting , classifying, and recovering potential resource errors leading to related QoS (constraints) conflicts and, eventually, related mission errors. AFTS model describes what and how many resources are required to perform a set of actions with the related QoS constraints to achieve missions.

The AFTS model description language consists of several specification components: *mission, action, resource*, *situation,* and *QoS constraint*. The *mission* component represents an application issued by a user. It consists of actions. An *action* component is a representation of an active function (or object method) triggered by a situation. It makes a situation change into a different situation. The *situation* is a precondition for the action fulfillment. In order to fulfill an action, the action uses one or more resources represented by *resource* components. A *QoS constraint* is given on one or more actions, or one or more QoS constraints are given on an action.

In our model a mission is a sequence of actions, each of which has one or more QoS constraints. The formal representation of missions and actions is given by situation-aware contract specification language SA-CSL [10] as follows:

```
Mission mission_1(object_1.action_1, object_2.action_2,…., object_n.action_n)
  Situation-Aware-Object {
    [Incoming][activate at situation_1] action_1()
      RequiredResources (resource_11(amount_11),…, resource_1p(amount_1p))
      WithQoSConstraint (QoSconstraint-list_1);
          ……
  } object_1 ;
  ....
  Situation-Aware-Object {
  ....
  } object_n ;
```

where `mission`$_1$ is a mission name which has actions `object`$_1$`.action`$_1$`,` `object`$_2$`.action`$_2$`,….,` `object`$_n$`.action`$_n$. Each `action`$_i$ is activated at `situation`$_i$ using resources `resource`$_{i1}$`,…..,` `resource`$_{ik}$ with necessary amounts `amount`$_{i1}$`,…..,` `amount`$_{1k}$,respectively. `[Incoming | Outgoing | Local | ClientServer]` are the four pre-tags of an action. Their meanings are as follows:

- `[Local]` means this action does not include any inter-device communication.
- `[ClientServer]` means this action is a ClientServer action. That is, this action has inter-device communication and it specifies the communication partner devices.
- `[Incoming]` and `[Outgoing]` are for peer-to-peer communication actions. A peer-to-peer communication action does not specify the communication partner devices. In this case, the communication is set up by matched situation. The tags indicate the direction of the communication: `[Incoming]` means the action receives message and `[Outgoing]` means the action sends message out.

The *QoS Constraints* are expressed using arithmetic, comparison and logic expressions like those in the typical programming language. The result of QoS constraint evaluation is Boolean. In the mission representation, QoS$_i$ can be represented with a combination of existing QoS constraints. The following represents a general QoS definition format and QoS Constraints.

```
QoS-Definition {                 QoS-Constraints {
  type1 QoSname1 {                 QoSconstraint1 = (QoSname1 op1 value1);
      variable1=expression1;       QoSconstraint2 = (QoSname2 op1 value2);
      ..............................   ..................................................
      variablem=expressionm;       QoSconstraintn = (QoSnamen opn valuen);
      return variablem           }
   }  .............
};
```

where *QoSname*$_1$ is a name of QoS variables which can be reused in QoS constraint expressions. It is represented by a set of expressions that consists of arithmetic, logic and comparison expressions. QoSconstraint$_i$ is a QoS constraint using QoSname$_i$, operator$_i$ (<, >, =>, <=. ==. <>), and value$_i$. It means that QoSname$_i$ should be satisfied with a condition expressed by operator$_i$ and value$_i$.

3.3 Adaptive QoS Management Algorithm

The QoS Management Agent monitors and analyzes dynamically-changing QoS requirements during mission fulfillment and identifies specified QoS constraint violations related to an action of a given mission before the action is performed. The following algorithm detects, classifies, and recovers the resource errors for adaptive QoS management. To ensure required reliability of multimedia communication systems, AFTS consists of 3 steps that are an error detection, an error classification, and an error recovery. The scheme of error detection is as follows. We are first in need of a method to detect an error for session's recovery. One of the methods to detect an error for session's recovery inspects PDB(process database) periodically. But this method has a weak point of inspecting all processes without regard to session. Therefore, we propose AFTS. This method detects an error by polling periodically the process with relation to session. Windows 95/98//XP creates a process database to represent the process. Process database include a list of threads, a list of loaded modules, the heap handle of the default process heap, a pointer to the process handle table, and a pointer to the memory context that the process runs in. A process handle is essentially the same thing as a file handle. GetExitCodeProcess() function retrieves the termination status of the process specified by the hProcess handle passed in. While a process is still actively running, its exit code is 0x103(0x: hexadecimal code). Second, FCA is an agent that plays a role as an interface to interact between FDA for detection and FRA for recovery. FCA has a function which classify the type of errors by using learning rules. FCA deals with learning in reactive multi-agent systems. Generally learning rules may be classified as supervised or unsupervised or reinforcement learning. Reinforcement learning is similar to supervised learning, except it, instead of being provided with the concept output for each network input, the algorithm is only given a grade. The grade(or score) is a measure of the network performance over some sequence of inputs[11]. This paper deals with Q-learning that is one of the reinforcement learning. Because FCA has not knowledge of error classification, it receives an acknowledgement information which is necessary for fault diagnosis from PDB(Process Data Base). Hence the training set consists of a set of input vectors, each with its desired target vector. Input vector components take on a continuous range of values. Target vector components valued. After training, the network accepts a set of continuous inputs and produces FCA can decide whether it is hardware error or software error based on learning rules. Third, after a system is detected, it processes recovery. First it is decided whether it is hardware error or software error. In case of software error, it can be recoverable. The scheme of error recovery method is different each other. It can be classified as many cases. In unrecoverable case, the system has to be restarted by manual when error occurred in hardware resources. In recoverable case, recoverable case classified as state insensitive and state sensitive. This approach has no consideration of domino effect between processes.

4 An Example of the AFTS Model

In this paper, the AFTS access requires situation-aware fault-tolerance QoS, in which the different fault-tolerance can be automatically enforced according to different situations such as wired or wireless network environment.

It represents an example of the AFTS model to support non-stop VOD service from situation 1 (Location = "on street" ^ Device = "handheld" ^ network = "wireless") into situation 2 (Location = "home" ^ Device = "TV" ^ network = "wired"). The VOD service is initiated by the *Wireless-VODservice* mission. At first, the actions, related with *Wireless-VODservice* mission such as A_{11}, A_{12}, etc, are triggered by *situation*$_1$. These actions make *VOD-server*$_1$ provide the VOD service to *VOD-client*$_1$ using the related resources with satisfying two constraints, *Fault-toleranceQoS*$_1$. The *Fault-toleranceQoS*$_1$ constraint is forced on Fault-Detection/ Fault-Recovery to execute detection at the *VOD-server*$_1$ before the VOD data transmission and to execute recovery at the *VOD-client*$_1$ just after receiving the VOD data. Since the *situation-aware, adaptive VOD service* mission is changed from wireless to wired network, *situation*$_2$ is created and the actions for *Wired-VODservice* are triggered. The *Fault-toleranceQoS*$_2$ constraints are enforced when the actions are executed. *VPN*$_0$ is executed on the *VOD-client*$_2$ to strengthen the fault-tolerance of the wired network. AFTS consist of FTA(Fault Tolerance Agent), UIA(User Interface Agent) and SMA(Session Management Agent). UIA is an agent which plays a role as an interface to interact between the user and FTA. UIA is a module in AFTS. UIA has functions which receive user's requirement and provides the results for the user. SMA is an agent which plays a role in connection of UIA and FTA as management for the whole information. SMA monitors the access to the session and controls the session. It has an object with a various information for each session and it also supports multitasking with this information. SMA consists of GSM(Global Session Manager), Daemon, LSM(Local Session Manager), PSM(Participant Session Manager), Session Monitor ,and Traffic Monitor. GSM has the function of controlling whole session when a number of sessions are open simultaneously. LSM manages only own session. For example, LSM is a lecture class in distributed multimedia environment. GSM can manage multiple LSM. Daemon is an object with services to create session. This system consists of a FTA, GSM, LSM, PSM and the application software on LAN. Platform 1 consists of GSM, Session Monitor, and Traffic Monitor. The other platform consists of Daemon, Local Session Manager, Participant Session Manager and FTA. Each platform except platform1 has a FTA. FTA is an agent that plays a role in detecting an error and recovering it. FTA informs SMA of the results of detected errors. Also, FTA activates a failure application software automatically. It informs SMA of the result again. FTA consists of FDA, FCA, and FRA. That is, FTA becomes aware of an error occurrence after it receives requirement of UIA. FDA has a function of error detection. FCA has a function of error classification. FRA has a function of error recovery. You can see message flows in FTE. It consists of Daemon, Session Manager and FTA. The relationship among FTA, Daemon and Session Manager are as shown in Figure 2 and Figure 3. The strong point of this system is to detect and recover an error automatically in case that the session's process come to an end from software error.

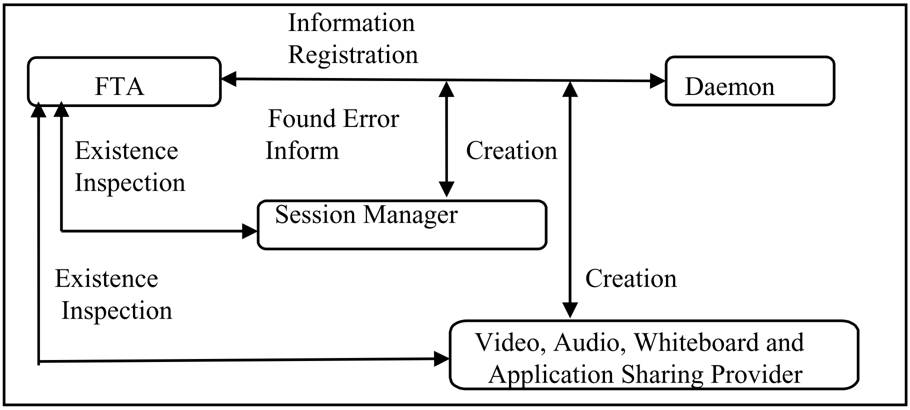


Fig. 2. Relationship between FTA and Daemon

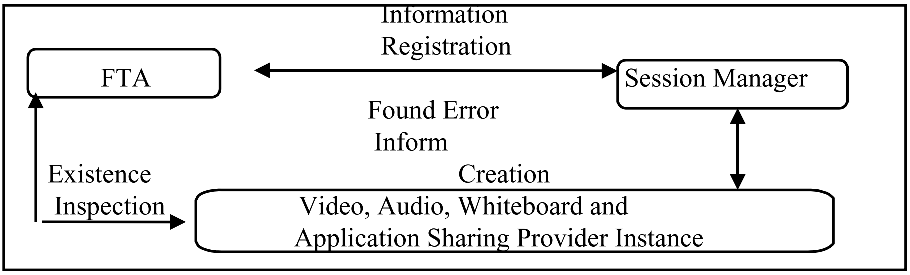


Fig. 3. Relationship between FTA and Session Manager

5 Related Work and Discussion

The focus of situation-aware ubiquitous computing has increased lately. An example of situation-aware applications is a multimedia education system. The development of multimedia computers and communication techniques has made it possible for a mind to be transmitted from a teacher to a student in distance environment. This paper proposes an Adaptive Fault Tolerance (AFT) algorithm in situation-aware middleware framework and presents its simulation model of AFT-based agents. FTE provide several functions and features capable of developing multimedia distant education system among students and teachers during lecture. AFT is a system that is suitable for detecting and recovering software error based on distributed multimedia education environment as FTE by using software techniques. This method detects an error by using process database. The purpose of this research is to return to a healthy state or at least an acceptable state for FTE session. It is to recover application software running on situation-aware ubiquitous computing automatically. When an error occurs, FTA inspects it by using API function for process database. If an error is found, FTA decides whether it is hardware error or software error. In case of software error, it can be recoverable. FTA informs Daemon and Session Manager of the fact. As they receive the information from the FTA, Daemon and Session Manager recovers from the error. The purpose of AFT system is to maintain and recover for FTE session automatically.

In the future work, fault-tolerance system will be generalized to be used in any environment, and we will progress the study of domino effect for distributed multimedia environment as an example of situation-aware applications.

Acknowledgement. I would like to thank Dr. Hoh Peter In for precious comments and suggestions.

References

[1] Mark Weiser, “The computer for the 21st century”, Scientific American, 265(30): 94-104, 1991.

[2] S. Yau, F. Karim, Y. Wang, B. Wang, and S. Gupta, "Reconfigurable Context-Sensitive Middleware for Pervasive Computing," *IEEE Pervasive Computing,* 1(3), July-September 2002, pp. 33-40.

[3] S. S. Yau and F. Karim, "Adaptive Middleware for Ubiquitous Computing Environments", *Design and Analysis of Distributed Embedded Systems, Proc. IFIP 17th WCC*, August 2002, Vol. 219, pp. 131-140.

[4] S. S. Yau and F. Karim, "Contention-Sensitive Middleware for Real-time Software in Ubiquitous Computing Environments", *Proc. 4th IEEE Int’l Symp. on Object-Oriented Real-time Distributed Computing (ISORC 2001)*, May 2001, pp. 163-170.

[5] D. Xu, et al., “QoS and Contention-Aware Muiti-Resource Reservation”, 9th IEEE International Symposium on High Performance Distributed Computing (HPDC’00), 2000.

[6] P. Bellavista, A. Corradi, and R. Montanari, “An Active Middleware to Control QoS Level of Multimedia Services”, Proceedings of the Eight IEEE Workshop on Future Trends of Distributed Computing System. 2001

[7] D. Xu, D. Wichadakul, and K. Nahrstedt, “Resource-Aware Configuration of Ubiquitous Multimedia Services”, Proceedings of IEEE International Conference on Multimedia and EXPO 2000(ICME 2000), 2000.

[8] K. Nahrstedt, D. Xu, and D. Wichadakul, “QoS-Aware Middleware for Ubiquitous and Heterogeneous Environments”, In IEEE Communications Magazine. 2001.

[9] J. Huang and Y. Wang, and F. Cao, “On Developing Distributed Middleware Services for QoS- and Criticality-Based Resource Negotiation and Adaptation”, Journal of Real-Time System, 1998.

[10] S. S. Yau, Y. Wang, D. Huang, and H. In “A Middleware Situation-Aware Contract Specification Language for Ubiquitous Computing”, submitted to FTDCS 2003.

[11] Martin T.Hagan, Howard B. Demuth, Mark Beale: Neural Network Design, PWS Publishing Company (1996) pp.4-3.

Design and Implementation of Middleware for Context-Aware Service Discovery in Ubiquitous Computing Environments*

Kyu Min Lee, Hyung-Jun Kim, Ho-Jin Shin, and Dong-Ryeol Shin

School of Information and Communication Engineering,
Sungkyunkwan University,
300 Cheoncheon-dong, Jangan-gu, Suwon, Gyeonggi-do 440-746, Korea
{kmlee, mairi, hjshin, drshin}@ece.skku.ac.kr

Abstract. The purpose of service discovery techniques is to minimize the cost of detecting services and provide users with convenience, even though various devices and services exist. For more dynamic and useful service discovery, middleware for context-aware service discovery is required. In this paper, a middleware system is designed and implemented, based on the agent platform for context-aware service discovery. When a service is detected, context information relating to the user and environment is used. As a policy-based system, our middleware does not only use context information, but also use predefined policy. In other words, user preference can be considered. Also, it has authentication module, so users having authority can only access the middleware. Near the conclusion of this paper, a hospital scenario is composed and implemented, by applying the proposed middleware solution.

1 Introduction

The compounding rate of technological sophistication is continually developing at high speed. These technologies represent increased convenience for consumers. For example, consumers can surf the Internet, conduct online shopping and control home devices, while walking outside, and using a handheld device, such as a PDA. Similar to this example, emerging ubiquitous and pervasive computing solutions provide "anytime, anywhere" computing, decoupling users from devices and viewing applications as entities for performing tasks on behalf of users.

The emerging availability of services in ubiquitous environments offers exciting new possibilities and challenges for both service users and service providers. Service providers are facing challenges involved in creating value-added services, by integrating information from various domains, while users of these services have the chance to access new services, therefore, countering information overload prevalent in modern society.

* This research is supported by the ubiquitous Autonomic Computing and Network Project, the Ministry of Information and Communication (MIC) 21st Century Frontier R&D Program in Korea.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 483 – 490, 2006.

Thus, both service providers and users of these services require a middleware system to integrate heterogeneous environments, employing various techniques to reduce complexity. There are many existing middleware system to solve that problems. Although they try to solve integration problems, another problem emerges. That is to use context information such as temperature, position and body information because service users can find more appropriate service if the middleware reflects context information, so recent middleware approach is to support context-awareness.

A number of context-aware systems have been developed, however, these systems have limits, for example complete service discovery in heterogeneous environments, and a no policy system to acquire individual user' requirements or users' subscriptions. Context-aware systems should be able to generate and handle various types of context in order to support users in ubiquitous environments.

In this paper, we design a middleware system based on Java Agent DEvelopment Framework (JADE) [1], a popular agent platform, for context-aware service discovery. In our middleware, raw context information is discarded and useful raw data is represented to high-level data. Context information can be combined, inferred by Context Manger which is comprised of several modules in our middleware. Because our middleware also have Policy Module, it can consider users' preference. Also, it has authentication module, so users having authority can only access the middleware. Finally, implementation is achieved in a hospital scenario.

The remainder of the paper is organized as follows. Section 2 describes of context and existing approaches. It presents the motivation for this paper, and describes the related work. Section 3 presents our proposed middleware for context-aware service discovery in detail. In section 4, we implement our middleware in a hospital scenario. Finally, the conclusion and future work are discussed in section 5.

2 Related Work

2.1 Definition of Context

Context refers to the meaning that can be inferred from the adjacent text, composed of con (with) and text. Dey considers context as any information regarding the circumstances, objects, or conditions by which a user is surrounded, and that is considered relevant to the interactions between the user and environment [2]. Many researchers have attempted to define context by enumerating examples. Schilit et al. divide context into three categories [3], 1) computing context such as network connectivity, communication costs, communication bandwidth, and nearby resources such as printers, displays, and workstations; 2) user context such as a user's profile, location, people nearby, and sometimes even current social situations; and 3) physical context such as lighting, noise levels, traffic conditions, and temperature.

2.2 Existing Approaches

This subsection presents the existing approaches that consider the contextual information in the service discovery process. The problems of using contextual information in these approaches are also discussed.

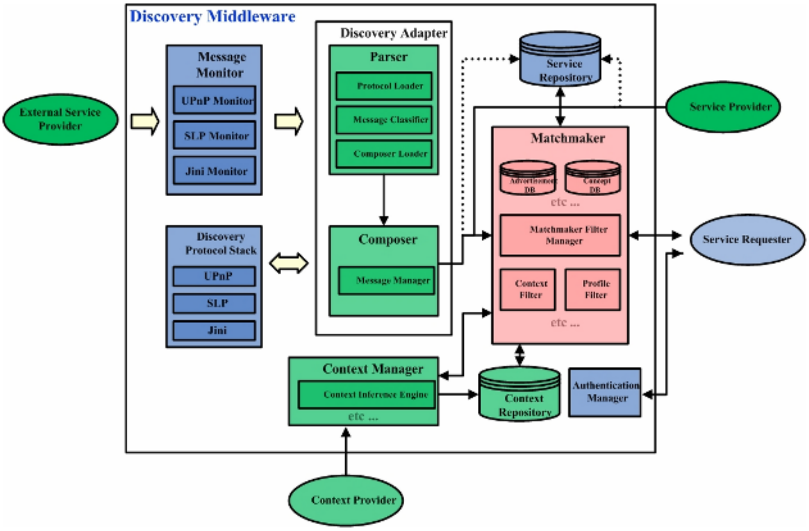


Fig. 1. The Architecture of Middleware

The *Cooltown* [4] project allows users to discover services that are in the user's vicinity. In this approach the location of the user and the service is used to conclude that the user is in a specific service area. In this way, services that are close to the user are returned by the service discovery mechanism. *The context toolkit* [5] represents a development toolkit that provides functionality to discover services using contextual information. It allows for describing services by means of white and yellow pages that include contextual information. *The platform for adaptive applications* [6] proposes architecture for applications that adapt their behavior according to the context of the user. The platform enables discovery of context providers using the type of context advertised. This contextual information is used to adapt application behavior. The *CB-Sec* project [7] provides functionality to discover services that are in the vicinity of the user. This approach takes into account the user and service capabilities in the service discovery process. *RCSM* [8] provides an object-based development framework for context-awareness. In order to support ad hoc communication, an Object Request Broker (ORB) is provided for communication transparency in a distributed environment. This includes monitoring and sensing functions.

As mentioned above, these projects attempt to consider users' context information when discovering services. While most of them also make dynamic and convenient systems for users, they only use some components of context information, and processing of raw context information is simple. These projects do not have any authentication module and policy module. In contrast to them, novel middleware is based on JADE, and have authentication module and policy module for context-aware service discovery. In the next section, the proposed middleware system is presented.

3 Proposed Middleware System

3.1 Architecture

Novel middleware is designed for context-aware service discovery. As demonstrated in Figure 1, this middleware consists of various modules such as parser, composer, service repository, and so on. In this paper, context-awareness in middleware is focused on, therefore other modules are discussed only briefly.

The Message Monitor has each discovery agent as a monitoring module for each service discovery protocol or network domain that can detect all messages from them because each network protocol or domain uses a well-known IP and Port number, i.e. 239.255.255.250:1900 on UPnP and 239.255.255.253:2427 on mSLP [9]. The Discovery Adapter consists of a Parser, which parses messages, and Composer, which composes indispensable data parsed from messages. It means the Composer represents the parsed data to the appropriate format, DAD (DF-Agent-Description) [1] format. The Composer then registers it with the Service Repository such as the Directory Facilitator (DF) in an agent platform [1]. The agent-based service provider can also register services directly via an Agent Communication Language (ACL) message [1]. When a service requester on behalf of a user requests a service, a request message is transmitted to the Matchmaker, which calculates which service is the most appropriate for the service request. The Context Provider such as light, camera and temperature, serves raw context information to the Context Manager. The Context Manager consists of modules which guarantee context-awareness. Useful context information is stored in the Context Repository. If you want see detail information about the middleware, you can see [10] [11]. In the next subsection, context modules are presented in detail.

3.2 Context Awareness

By introducing modules in Context Manager, people can know how Context Manager is comprised of. The Context Manager consists of six modules and one database as shown in Figure 2. Each of them can be described as the following:

*** Context Device Manager**

In ubiquitous environments, there are many diverse context devices and context sources. They frequently enter or go out in the network, so we design the Context Device Manager, which manages them. When they register or deregister, a message is sent to the Context Device Manager. Then, it recognizes the context sources composed in the middleware network.

*** Context Filtering Module**

The next module is the Context Filtering Module. Its main purpose is to protect the Context Repository from being flooded with excessive information. When context information (e.g., time) from context sources is continuously delivered, users or user agents may be interested in receiving a few values periodically. This is especially important for both the Matchmaker and users because the Matchmaker can search requested context information in the Context Repository as soon as possible and users reduce the response time of the Matchmaker.

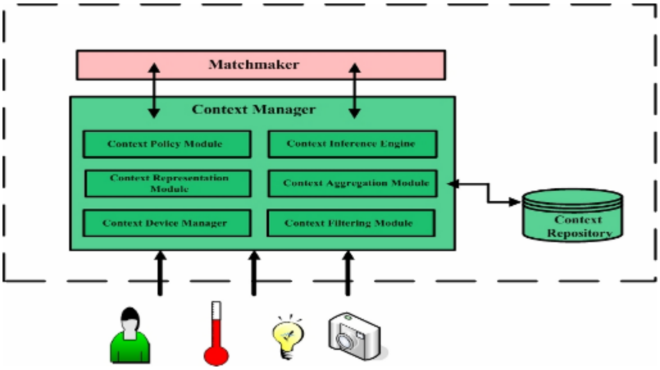


Fig. 2. Context Modules

* Context Representation Module

The Context Representation Module converts raw context information from context sources such as sensors into normalized forms. Information from various context sources, in general, exist in ubiquitous computing environments and there are many different types of data, e.g., binary, integer, real number, etc. For the more, the representation may be different from each other. This will result in difficulties not only in usability, but also in production of higher-level contexts from lower-level contexts. This module represents raw data in normalized forms.

* Context Aggregation Module

The Context Aggregation Module has the ability to combine the context information from the Context Representation Module. This module produces combined contexts. The combined context can be used to provide better information with users because they can understand easily what kind of situation. For example, when Context Representation Module generates Location (“713”), Patient (“Chris”), Sensor (“Bodily Temperature”), Temperature (“34”), Context Aggregation Module creates Room(Number, “713”, Patient, “Chris”, Sensor, “Bodily Temperature”, Temperature, “34”).

* Context Inference Module

Context Inference Module can contain one or more context reasoning modules, based on the users’ requirements. The need for a Context Inference Module arises because not all information can be gathered from context sources. It is used to derive higher-level context information from lower-level information. Therefore, the module infers new context information from the current context. This makes the middleware system increasingly intelligent. For example, in a room a man lies down on a bed, he turns off lights and closes his eyes. This means he is sleeping.

* Context Policy Module

When a situation or a environment is configured, users might want to know the moment. At that time, middleware invokes actions already described. This is possible due to the Context Policy Module. User can send policy message as an ACL message format to the Context Agent, which has Context Policy Module. In other words, this invokes actions if a pre-defined context is detected in the current situations or surroundings.

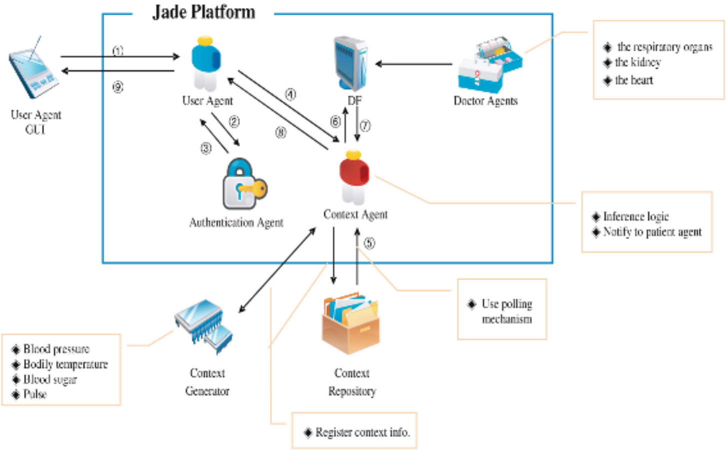


Fig. 3. Contest-aware Service Discovery Scenario

4 Implementation

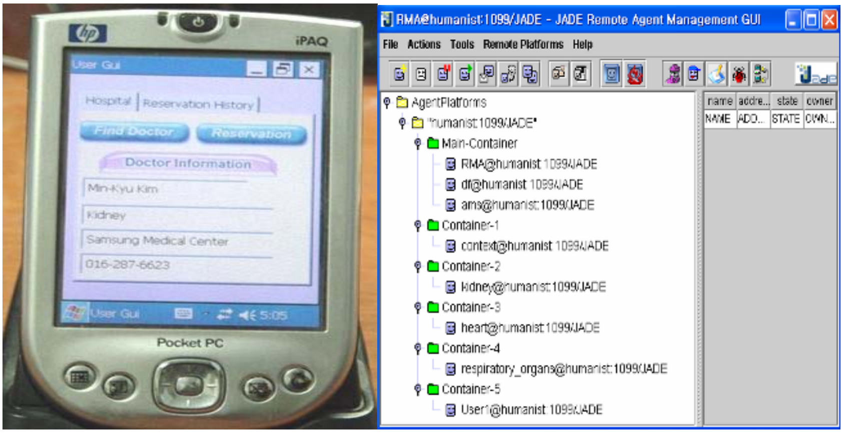
After designing and implementing middleware, the best way to test the middleware is to create a scenario, therefore, in this paper a hospital scenario is used. As shown in Figure 3, the scenario is described as the following:

First, we create three doctor agents such as the respiratory organs, the kidney and the heart, service providers in service discovery mechanism. Next, we register them to DF on JADE and create 4 sensors such as blood pressure, bodily temperature, blood sugar, and pulse as context sources. We then build the context repository as a database. In this scenario, doctor agents mean medical specialists of each medical field.

Context information is generated by the context generator and is filtered, represented and aggregated by the context agent, the Context Manager described in subsection 3.2. The context information is then stored in the context repository. In fact, service discovery is started by a user agent on behalf of a user. It send identification message to authentication agent with Authentication Manager ①②, and it then receives permission message from authentication agent if the user has authority to access middleware ③.

After that, it asks the context agent to discover the most appropriate service with considering bodily context information ④⑤. If the user' body condition is normal, 'normal' will be presented in the user's GUI. Otherwise, the context agent finds a suitable doctor ⑥⑦. As an example, the user's GUI shows the doctor's profile and other information in Figure 4 ⑧⑨. Now, the user is able to contact the doctor as soon as possible. We can also see existing agents in the middleware in Figure 4.

The user agent is implemented using Personal Java 1.2, J2ME CLDC/MIDP and JADE-LEAP. For this implementation, we use an IBM thinkpad and WinCE based



Min-Kyu Kim
Kidney
Samsung Medical Center
016-287-6623

```
AgentPlatforms
  "humanist:1099/JADE"
    Main-Container
      RMA@humanist:1099/JADE
      df@humanist:1099/JADE
      ams@humanist:1099/JADE
    Container-1
      context@humanist:1099/JADE
    Container-2
      kidney@humanist:1099/JADE
    Container-3
      heart@humanist:1099/JADE
    Container-4
      respiratory_organs@humanist:1099/JADE
    Container-5
      User1@humanist:1099/JADE
```

name	addre...	state	owner
NAME	ADD...	STATE	OWN...

Fig. 4. User and JADE RMA GUI

iPAQ's from Compaq. The middleware is implemented using J2SE 1.4 and JADE 3.3 in a Desktop computer. This middleware is only a preliminary version, and we are now in the process of completing the implementation.

5 Conclusion and Future Work

This paper presents middleware based on JADE for context-aware service discovery. The use of it offers scalability support, which means users can use various services existing in different network domains. If users use the proposed middleware in the home, they can control devices. Additionally, context-awareness is guaranteed via several context modules. In other words, various types of context can easily be handled with representation, aggregation and inference, in order to provide appropriate services to users in ubiquitous environments. The key point of the proposed middleware is the Policy Module which considers users' preference, through which they can subscribe when a particular situation is made. Also, users having authority can only access the middleware because it has Authentication Manager.

We are currently developing a more sophisticated version of our middleware, while simultaneously performing simulations and evaluating the performance of the existing system. Finally, we will expand our middleware system, using ontology in a part of agent communication and build ontology database as future works because users can search service semantically through ontology. It will be suitable middleware for a ubiquitous environment.

References

1. JADE, http://jade.tilab.com, Telecom Italia Lab, 2005
2. A. K. Dey, G. D. Abowd and D. Salber, "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications", Human Computer Interaction Journal, Special Issue on Context-Aware Computing, Volume: 16, Issue: 1, 2001.

3. B. Schilit, N. Adams and R. Want, “Context-Aware Computing Applications”, Proc. of The IEEE Workshop on Mobile Computing Systems and Applications, Santa Cruz, California, USA, 1994.
4. Cooltown project, http://www.cooltown.com/cooltown/index.asp, Hewlet Packard, 2004.
5. The context toolkit, http://www.cs.berkeley.edu/~dey/context.html , Dey, A., 2004.
6. C. Efstratiou, K. Cheverst, N. Davies and A. Friday, “An architecture for the Effective Support of Adaptive Context-Aware Applications”, Mobile Data Management, Pages: 15-26, 2001.
7. S. Kouadri Mostefaoui, A. Tafat-Bouzid, and B. Hirsbrunner, “Using Context Information for Service Discovery and Composition”, Proc. of the 5th Conf. on information integration and web-based applications and services, Jakarta, Pages: 129-138, 2003.
8. S. S. Yau and F. Karim, “An Adaptive Middleware form Context-Sensitive Communications of Real-time Applications in Ubiquitous Computing Enviornments,” Real-Time Systems, Volume: 26, Pages: 29-61, 2004.
9. Universal Plug and Play Specification, v1.0, http://www.upnp.org.
10. Kee-Hyun Choi, Kyu Min Lee, Ho-Jin Shin and Dong-Ryeol Shin, "Efficient Algorithm for Service Matchmaking in Ubiquitous Environments", Proc. of the Springer on DEXA (EC-Web 2005), Pages: 258-266, August 2005.
11. Hyung-Jun Kim, Kyu Min Lee, Kee-Hyun Choi and Dong-Ryeol Shin, “Service Discovery using FIPA-Compliant AP to Support Scalability in Ubiquitous Environments”, Proc. of the IEEE on ICIS, Pages: 647-652, July 2005.

A Dynamic Channel Allocation Mechanism in Cellular Mobile Networks for Ubiquitous Environments Based on Time Constraints

SeongHoon Lee[1], DongWoo Lee[2], Donghee Shim[3], Dongyoung Cho[3], and Wankwon Lee[3]

[1] Division of Information & Communication Engineering,
Cheonan University, 115, Anseo-dong, Cheonan,
Choongnam, Republic of Korea
shlee@cheonan.ac.kr
[2] Department of Computer Engineering,
Woosong University, 17-2, Jayang-dong, Dong-ku,
Daejeon, Republic of Korea
dwlee@woosong.ac.kr
[3] School of Information Technology & Engineering, Jeonju University,
1200, 3rd Street Hyoja-dong Wansan-Koo, Jeon-Ju, Chonbuk, Republic of Korea
{dhshim, chody, wklee}@jj.ac.kr

Abstract. The new real-time applications like multimedia and real-time services in a wireless network for ubiquitous environments will be dramatically increased. However, many real-time services of mobile hosts in a cell cannot be continued because of insufficiency of useful channels. Conventional channel assignment approaches didn't properly consider the problem to serve real-time applications in a cell. This paper proposes a new real-time channel assignment algorithm based on time constraint analysis of channel requests. The proposed algorithm dynamically borrows available channels from neighboring cells. It also supports a smooth handoff which continuously serves real-time applications of the mobile hosts.

1 Introduction

The new mobile applications like multimedia and real-time services in a wireless network will be dramatically increased for ubiquitous environments in the future. So the frequencies called channels should be provided for many mobile hosts. However, wireless channels are scarce resources in current mobile services. Therefore the frequency reuse is the main issue to increase system capacity of a wireless telephone system. The area in a wireless network to be served is divided into several regions, called cell. Each cell has a Base Station (BS), and the base station maintains a wireless link with mobile users.

The classical method for reusing a limited set of channels is Fixed Channel Allocation (FCA) algorithm[1, 2, 3]. In this scheme, each BS is allocated a fixed set of channels to use, and the allocation is done in a manner that the distance between cells using the same channels is large enough to avoid interference between users in different cells using the same channels[4]. This scheme works reasonably with the

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 491 – 498, 2006.

uniformed distribution of users, but fails to adjust for the variation in the number of users in different cells. The conventional propagation models, on which the FCA related algorithms are based on pre-assignment of channels, appear to be totally invalid for the small size cells[5]. And the FCA methods are inappropriate to the hot cell problem which dynamically happens the overloaded channel requests onto the specific cell at any time[6]. To accommodate this situation, a variety of Dynamic Channel Allocation (DCA) algorithms have been proposed based on different methods[7, 8, 9]. Graph theoretic scheme was employed in[7], probabilistic model in[8], and heuristic methods in [9].

However these papers are not appropriate to perform a real-time channel borrowing strategy because the real calls in each cell were not considered for time constraint to the deadline. Unlike other traditional works, our algorithm employs a new Real-time Dynamic Channel Allocation (RDCA) method for real-time traffic services and shows more efficiency in the dynamic use of available channels at any time. We propose a new scheme which identifies and maintains the call patterns for every σ unit times through the time constraint analysis of call requests in cells. The call requests of mobile hosts in every cell are collected every period (σ unit times) and maintained into the list of call requests called $CList_i$, which indicates the summation of call requests of i_{th} period in a cellular system. And channel borrowers borrow channels and lenders lend channels using information of time to deadline of call requests in the call list $CList_i$. The channel borrowing process also supports smooth-handoff scheme. If a mobile host can continue using the same channel in the new cell, it does not have to retune to a new channel after handoff. Our algorithm keeps the current allocated channels of handoff hosts as continuously as possible when a mobile host crosses over from one cell to another.

2 Channel Borrowing Strategy Based on Time Constraints

A real-time service must be arrived at the destination within its deadline, but a server fails to deliver a real-time service if the service arrives after its deadline. If a real-time service reaches at the goal before its deadline, the server needs to get the proper size of buffer to accept its calls. This paper assumes that the buffer size is infinite. The goal of this paper is to minimize the expected number of failed services over an infinite horizon. The presented algorithm logically organizes wireless environments and assumes several fundamental models as following subsections.

2.1 System Model

The used models in our cellular architecture are as follows. A geographical area consists of a number of hexagonal cells, each cell served by a BS. A group of cells are again served by a Mobile Switching Center (MSC). The MSCs are connected through a wired network, and each MSC also acts as a gateway and channel server for the wireless networks to the BSs. Each cell previously has a fixed set of N_c channels according to the compact pattern based on the fixed assignment scheme[1]. The value of N_c assigned in each cell is seven. A compact pattern with shift parameters of i=3 and j=2 is used in this paper. Two parameters i and j are called shift parameters. The number of cells, *N* in a compact pattern is given by following equation, which means the number of different channel sets[10].

$$N = i^2 + i \times j + j^2 \tag{1}$$

Fig 1 shows a cellular system based on compact patterns which are applied to our algorithm.

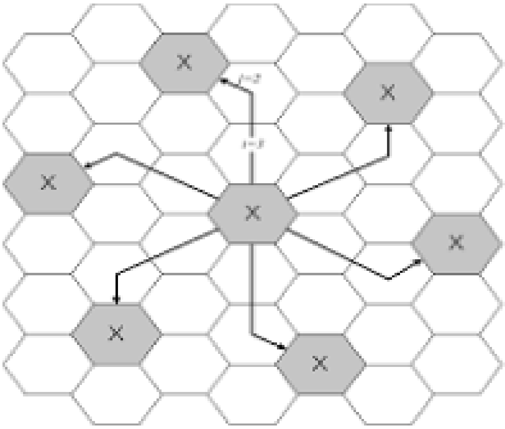


Fig. 1. Cellular System based on Compact patterns

A centralized channel server dynamically services call requests in every cells. The channel server maintains a channel pool which is a set of channels in its cellular system. When calls require channels for server, the calls inserted into call entry $CList_i$ and periodically rearranged by the order of the nearest to the deadline within such a list. If there are two or more calls with both the same time constraint and the different call types in $CList_i$, those calls are rearranged according to call types of following sequences: handoff call, new call, and termination(or close) call. At initial state, the server uniformly assigns the same number of channels to each cell.

A cell can be evaluated according to the value(δ_h) of its degree of hotness defined by Equation(2). If δ_h of a cell is greater than that of others, it means that the cells with the higher value have more calls. δ_h has the range of $0 \leq \delta_h \leq 1$.

$$\delta_h = \textit{numbers of allocated channels} / N_c \tag{2}$$

The mobile users in wireless networks can be classified into six categories: staying user with channel requests (new call), incoming handoff user with channel requests (open handoff call), departing handoff user with returning channels (close handoff call), ending user with returning channels (close call), staying user with using channels (old call) and the user walking about cells without using channels (roaming call). The proposed algorithm only performs channel services, which allocate channels to new call and open handoff call, and recollect close call and close handoff call.

This model assumes that each call requires a unit frequency for a channel server and has time constraint information about the remaining time to the deadline. Also, the following assumptions are made: calls at any cell arrive according to Poisson process, λ; channel service time is exponentially distributed with mean length $1/\mu$; the time constraint of calls as real-time parameter has the random value within the range from the current accepted time of the call to the blocked/dropped time (or during the call duration). In our system model, the handoff users which across from a cell to the other arrive as Poisson process.

2.2 Real-Time Dynamic Channel Allocation

A real-time channel borrowing algorithm can be defined by the relationship between borrower and lender. When a BS with calls requiring for channel, it becomes a borrower B. When a BS with lending channels, it becomes a lender L. The algorithm borrows channels dynamically using following parameters in each cell.

□ ***hotness*****:** The ratio of the number of allocated channels in a cell L to the total number of channels allocated determines the degree of hotness, δ_h(L), of that cell with a minimal allocated channel.

□ ***deadline nearness*****:** The temporal distances by unit time until calls of the lender L with available channels is arrived %into service deadline within its time information to deadline of $CList_i$.

□ ***spatial nearness*****:** It represents the spatial cell-distance $D_s(B, L)$between the borrower B and lender L in a compact pattern.

□ ***available condition*****:** The number of available channels of non-co-channel cells of the lender cells L which are also non-co-channel cells of the borrower cell B in a compact pattern, is denoted by $A_c(B, L)$. Since our cellular architecture has a compact pattern with nineteen cells as Equation(1) and each cell with seven channels, the number of channels in non-co-channel cells of the lender cell L satisfies the condition of $0 \leq A_c(B, L) \leq 133$.

To allocate channels effectively, the channel borrowing strategy selects the cell whose the value of cost function $C_f(B, L)$ is maximum among cells as lender.

$$C_f(B, L) = A_c(B, L) /(((D_s(B, L)/R_{cp})\times(\delta_h (L))) \tag{3}$$

where R_{cp} denotes the radius of the compact pattern in terms of cell distance which implies $1 \leq D_s(B, L \leq R_{cp}$. The proposed channel allocation scheme for cellular mobile environment is centralized in nature because it is applied to a few cells. This implies that the load on the central server would not be too high.

The proposed channel allocation algorithm including processes above is shown such as following step by step. The channel allocation steps in the algorithm employ $CList_i$, the type of call (new, handoff or close call) and cost function.

Call Arrival: When a call arrives, the algorithm evaluates the cost function $C_f(B, L)$ for each cell with free channels and assigns the channel that leads to the cost function with the largest estimated value. If no free channel is currently available, the call must be blocked.

Call Handoff: When a mobile user across from a cell to others, the call is handoff call to the cell of entry; that is, if there is the same free channel used by the handoff call, the same channel is provided to the call at the new cell. Otherwise, a new free channel

is provided to it. If no such channel is available, the call must be dropped from the system.
Call Termination: When a call terminates, one by one each ongoing call in that cell is considered for reassignment to the just freed channel; the results of cost function $C_f(B, L)$ are evaluated and compared to the value of not doing any reassignment at all. The action that leads to the highest value of is then executed.

The *Phase I* of our algorithm can be initiated periodically by BS and *Phase II* periodically by BS whenever a borrower cell requests channels to MSC as channel server. Every BS is responsible for updating the parameters of server such as the current degree of hotness δ_h of the corresponding cells.

[Phase I] Operations at Cells :

- □ *Step 1*: Each BS updates its hotness δ_h on server a period.
- □ *Step 2*: When a MH begins its handoff service, starts or terminates its service, its BS sends the identifier, time constraint and other information of the MH and the identifiers of channels occupied by the MH to MSC.

[Phase II] Operations at Channel Server :

- □ *Step 1*: MSC periodically manages $CList_i$ including all calls from BSs. Firstly, all calls are arranged in the order of the earliest deadline parameter. Secondly, in the case of the same deadline values, in the order of the call types as handoff, new and termination call. Following steps focus on channel allocation for the handoff and new calls.
- □ *Step 2*: MSC extracts a call from $CList_i$. Firstly, if the call is a handoff call, it collects the used channel from departing cell and allocates the same channel of incoming cell to MH for smooth-handoff if possible. Secondly, If the call is a new call, MSC gives a available channel of the corresponding cell. Finally, If the call is close call, MSC withdraws the used channel from the corresponding cell. If the call didn't have available channels yet, its BS as a borrower sends a borrowing message.
- □ *Step 3*: MSC receives a channel borrowing message from BS, it selects lender which maximizes the value of Equation(3) and transmits a borrowing message to the selected lender.
- □ *Step 4*: If the lender has a set of available channels, it locks lending channels and returns a acknowledgement message including lending channels to MSC. Otherwise, it returns a null message to MSC.
- □ *Step 5*: If MSC receives a set of available channels from lender, it sends a lending message to borrower. Otherwise, MSC blocks the call and goes on to Step 7.
- □ *Step 6*: The borrower receives a lending message from MSC. The borrowing channel is allocated to the call requiring channel.
- □ *Step 7*: Repeat each step after step 2 until all channel requests are satisfied or MSC has no cold cells with a set of available channels.

3 Simulation and Results

We simulated the proposed algorithm and some previous algorithms in environments with Intel Pentium-133 CPU and 32MB main memory. All programs were written using Java language. Our simulation results represent more efficient uses of available channels and more successful services of real-time channel requests over real-time applications in cellular networks. This method using the real-time channel request consideration minimizes the number of service blocking as compared to other papers. Fig 2 and Fig 3 show service failure rates in the cases of 150calls/hr and 200calls/hr. The cellular system consists of forty-nine cells, and each cell has seven channels as initial assignment.

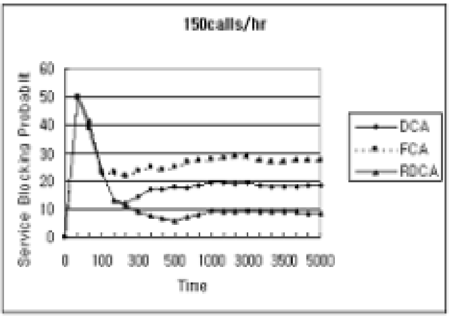


Fig. 2. Results with 150calls/hr

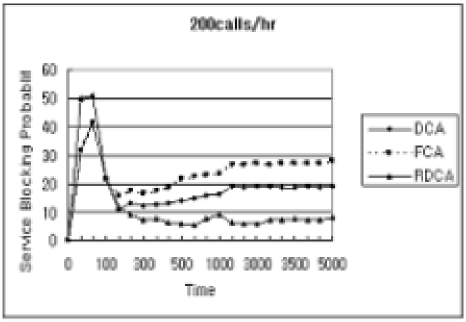


Fig. 3. Results with 200calls/hr

In these figures, the curves of our RDCA method are totally low relative to those of FCA and DCA of traditional methods since RDCA reorders channel requests in its call entry *CList*$_i$ in the order of the earliest deadline. All simulations start with no ongoing calls and therefore the blocking probabilities are high and low in the early minutes of the performance curves. If we give more channels to our all simulations, service blocking probabilities of all algorithms will be less than current results.

The simulation results of the proposed algorithm don't show the results of call blocking ratios because of the limited space. But they also display call blocking curves similar to service blocking probabilities in Fig 2 and Fig 3 if they know transfer speed and packet size of transfer data.

4 Conclusions

The proposed RDCA algorithm shows good experimental results since it firstly services the call with the earliest deadline and gives a new issue about wireless real-time and multimedia applications. The algorithm maintains the ordered call list with channel requests through real-time analysis in each cell. It gives new real-time channel allocation because it provides a real-time traffic service by time to deadline stored in system. This channel service minimizes the number of channel a service blocking ratio distinguishably since it firstly provide a free channel for the call request of the earliest deadline.

Future works will provide more advanced real-time allocation algorithms which efficiently execute channel services using the fixed or limited buffer size and consider admission control. We will explore the effort of decentralized approach in future work.

References

[1] V. H. Macdonald,``Advanced Mobile Phone Service: The Cellular Concept," Bell Systems Technical Journal**,** vol. 58, January 1996, pp. 15-41.

[2] S. M. Elnoubi, R. Singh, S. C. Gupta, ``A New Frequency Channel Assignment in High Capacity Mobile Communication Systems," IEEE Trans. Vehicular Technology, vol. VT-31, no. 3, August 1982.

[3] F. J. J. Romero and D. M. Rodriguez, ``Channel Assignment in Cellular Systems Using Genetic Algorithms," IEEE Vehicular Technology Conference, vol. 2, 1996, pp. 741-745.

[4] E. Linskog, ``Combating Co-Channel Interferers in a TDMA System Using Interference Estimates From Adjacent Frames," Proceedings of 29th Asilomar Conference on Signals, Systems, \& Computers, Pacific Grove, California, 1995.

[5] P. Harley, ``Short Distance Attenuation Measurements at 900MHZ and 1.8GHZ using Low Antenna Heights for Micro-cells," IEEE Journal on Selected Areas in Communications**,** vol. 7, no.1, January 1989, pp. 5-11.

[6] H. Jiang and S. S. Rappaport, ``CBWL: A New Channel Assignment and Sharing Method for Cellular Communication Systems," IEEE Trans. Vehicular Technology, vol. 43, no. 2, May 1994.

[7] H. C. Tan and M. K. Gurcan, ``A Fast Dynamic Channel Allocation Scheme for a Centrally Controlled Radio Local Area Network," IEEE Vehicular Technology Conference, vol. 2, 1996, pp. 731-735.

[8] S. K. Das, S. K. Sen and R. Jayaram, ``A Dynamic Load Balancing Strategy for Channel Assignment using Selective Borrowing in Cellular Mobile Environment," ACM/Baltzer Wireless Networks (WINET) journal, special issue on Mobicom'96, May 1997, pp. 333-347.

[9] S. Singh and D. Bertsekas, ``Reinforcement Learning for Dynamic Channel Allocation in Cellular Telephone Systems," NIPS'96**,** 1996:
http://www.cs.colorado.edu /~baveja/papers.html.

[10] W. C. Y. Lee, ``Mobile Cellular Telecommunication Systems, Analog and Digital Systems," Second Edition, McGraw-Hill, 1996.

[11] P. Mirchandani and Z. Xu, ``Performance Analysis of Integrated Voice/Data Communication in Cellular Systems with Virtually Fixed Channel Assignment," IEEE IPCCC'93, 1993, pp. 370-375.

[12] E. J. Wilmes and K. T. Erickson, ``Two Methods of Neural Network Controlled Dynamic Channel Allocation for Mobile Radio Systems," IEEE Vehicular Technology Conference, vol. 2, 1996, pp. 746-750.

[13] A. Merchant, D. Raychaudhuri, Q. Ren and B. Sengupta, "A Global Dynamic Channel Allocation Algorithm in Wireless Communications," IEEE GLOBECOM'97, vol. 2, 1997, pp. 1016-1021.

[14] S. K. Das, S. K. Sen, R. Jayaram and P. Agrawal, "An Efficient Distributed Channel Management Algorithm for Cellular Mobile Networks," Proceedings of IEEE International Conference on Universal Personal Communications (ICUPC), San Diego, California, Oct. 1997.

[15] C. Perkins, "Mobile IP", IEEE Communications Magazine, May 2002.

[16] R. Koodli, "Fast Handovers dor Mobile Ipv6", Internet-Draft, Draft-ietf-mobileip-fast-mipv6-08, October 2003.

Performance Analysis of Task Schedulers in Operating Systems for Wireless Sensor Networks*

Sangho Yi[1], Hong Min[1], Junyoung Heo[1], Boncheol Gu[1], Yookun Cho[1],
Jiman Hong[2,**], Jinwon Kim[3], Kwangyong Lee[3], and Seungmin Park[3]

[1] System Software Research Laboratory,
School of Computer Science and Engineering, Seoul National University,
San 56-1, Sillim-dong, Gwanak-gu, Seoul, 151-742, Korea
{shyi, hmin, jyheo, bcgu, cho}@ssrnet.snu.ac.kr
[2] School of Computer Science and Engineering, Kwangwoon University,
447-1, Wolgye-dong, Nowon-gu, Seoul, 139-701, Korea
gman@daisy.kw.ac.kr
[3] Ubiquitous Computing Middleware Research Team,
Electronics and Telecommunications Research Institute,
161 Gajeong-Dong, Yuseong-gu, Daejeon, 305-350, Korea
{john2004, kylee, minpark}@etri.re.kr

Abstract. In wireless sensor networks, power is a critical resource in battery powered sensor nodes. In this respect, as it is important to efficiently utilize the limited battery power, it would be desirable to make such nodes as energy efficient as possible. Many researchers who develop operating systems of wireless sensor networks have been trying to find a way to enhance energy efficiency of sensor nodes. In this paper, we present an overview of sensor node operating systems and some of its functionalities, and then present a performance analysis of task schedulers and task-related kernel routines of existing sensor node operating systems. The results of performance analysis show some advantages and disadvantages of the existing operating systems, and based on these information, we present some possible improvements for increasing the efficiency of sensor node operating systems.

1 Introduction

Nowadays, wireless sensor networks have drawn great attention as a new and important research area[1]. These sensor networks typically consist of hundreds or even thousands of sensor nodes deployed in a geographical region to sense events. They are used in many applications such as environmental control, offices, robot control, and automatic manufacturing environments, and moreover can be used

* The present research was conducted by the Research Grant of Kwangwoon University in 2006, and was supported in part by the Brain Korea 21 project.

** Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 499–508, 2006.

even in harsh environments[2, 3]. Developing wireless sensor networks entails significant technical challenges due to the many environmental constraints.

Recent advancement in sensor technology such as low power electronics, and low power RF design have made it possible to develop relatively inexpensive and low-powered multi-functional tiny sensor nodes[4]. These sensor nodes, which consist of sensing, data processing, and communicating components, leverage the idea of sensor networks. Such sensor networks are applicable to home networks, natural environments, health care, telematics systems, and many kinds of embedded systems[5, 6, 7, 8, 9]. For example, sensor nodes are deployed to gather information of the natural environments, for example, inaccessible mountains and even in harsh environments[6, 7, 8], and in telematics systems for sensing data to service the remote clients[9].

In wireless sensor networks, power is a critical resource in battery-powered sensor nodes. In this regard, efficient utilization of limited battery power is an important issue. Thus, it is desirable to make such nodes as energy efficient as possible. Today, many research on wireless sensor networks have been studied for improving the cost and energy efficiency of sensor nodes[1, 4, 10, 11, 12, 13]. For example, in [11, 13], the authors proposed a fundamental frameworks for sensor systems such as *Berkeley's TinyOS* sensor architecture, and also support the prototyping, programming, testing, and deployment of sensor networking applications. Another example is the *MANTIS* operating system which was proposed by [1] to meet the demands of advanced multimodal sensor networks deployments. Then, *SOS* operating system[10] was designed to improve energy efficiency of *TinyOS* and to support the idea of dynamic reprogramming in wireless sensor nodes. Also in [4], Lee et al. proposed a scalable and reconfigurable *Nano-Qplus* operating system and developed its sensor node platform architecture. These systems offer an integrated combination of general-purpose hardware platforms, and open-source embedded operating systems, a uniform API, system management and development tools.

In this paper, we present an overview of sensor node operating systems and its own functionalities, and show a performance analysis of task schedulers and task-related jobs in previous sensor node operating systems[1, 4, 10, 11]. The results of the performance analysis will demonstrate some of the advantages and disadvantages of the existing operating systems. These information will then be used to present some possible improvements for increasing the efficiency of sensor node operating systems.

The rest of this paper is organized as follows. In Section 2, we present related work on sensor node operating systems. Section 3 describes an overview of operating systems for sensor nodes. Section 4 presents and evaluates the performance of the sensor node operating systems, and present some possible improvements for these operating systems. Finally, conclusions are made in Section 5.

2 Related Work

In this section, we present a brief overview of the related work that have been done on sensor node operating systems. Considerable research efforts[1, 4, 10, 11, 12] have been made to improve the efficiency and extending functionalities of the wireless sensor networks and the sensor node operating systems.

In [11], Levis et al. proposed *Berkeley's TinyOS* architecture designs and implementations. *TinyOS* is a well-known operating systems and the *Mote* platform have been widely used in many kinds of applications[14, 15]. It is currently a fundamental framework of research on wireless sensor networks. In [12], Levis and Culler proposed a *Maté* on *TinyOS* architecture. *Maté* is a tiny communication-centric virtual machine designed for sensor networks. *Maté*'s high-level interface makes complex programs to be very short, hencd reducing the amount of energy consumed during transmission of new programs.

In [10], Han et al. proposed a *SOS* operating system, that consists of dynamically loadable modules and a kernel, which implements messaging, dynamic memory, and module loading and unloading, among other services. In *SOS*, modules are not processes. They are scheduled cooperatively. Individual modules can be added and removed with minimal system interruption.

In [1], Bhatti et al. proposed a *MANTIS* operating system. It supports preemptive multithreading, so it can handle multimodal sensors effectively. In addition, it enables micro sensor nodes to natively interleave complex tasks with time-sensitive tasks.

In [4], Lee et al. proposed a *Nano-Qplus* operating system for wireless sensor networks. *Nano-Qplus* is a scalable and reconfigurable operating system. It supports flexible *Nano-HAL(hardware abstraction layer)*, so the developers can work easily under various sensor nodes platforms. Furthermore, it supports effective power management mode and preemptive task schedulers so that the multimodal sensing jobs can be scheduled while at the same time keeping track of the energy efficiency of sensor nodes.

3 Overview of Operating Systems for Sensor Nodes

In this section, we present an overview of the operating systems for sensor nodes in wireless sensor networks and show some essential requirements for these operating systems.

3.1 Preliminaries on Wireless Sensor Networks

Figure 1 shows an example of the distribution of wireless sensor networks. Wireless sensor networks typically consist of hundreds or even thousands of sensor nodes deployed in a geographical region to sense events. There are challenges to designing an operating system for wireless sensor networks. Since it is essential that the cost-efficiency of producting sensor nodes, the sensor nodes, such restrictions limits memory space and energy lifetime. For example, *Berkeley's MICA* motes series have only 4 Kbytes run-time memory spaces, and its power sources are only 2 × *AA* batteries[16]. The following are essential factors that should be considered during the design process of a sensor node operating systems for wireless sensor networks.

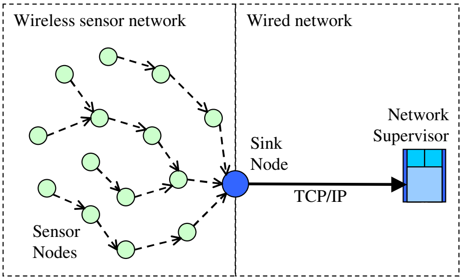


Fig. 1. An example of wireless sensor networks

- *Energy efficiency*
- *Tiny memory space*
- *Multimodal sensor support*
- *Dynamic reprogramming*
- *Real-time job processing*

In wireless sensor networks, *Energy efficiency* is an important issue, and thus, in general, sensor nodes have multimodal sensing devices(eg. temperature, humidity, light, and sound), thus *Multimodal sensor support* also becomes essential in sensor node operating systems. Furthermore, sensor node programs need recompiling or reprogramming in run-time. Therefore, *Dynamic reprogramming* is necessary. Finally, the sensor networks must operate under real-time constraint. Thus, *Real-time job processing* is also an important consideration in designing a sensor node operating system. Table 1 shows the comparison of operating systems for wireless sensor networks in those essential features[4].

Table 1 shows the *pros* and *cons* of four sensor node operating systems. These functional and featural difference are explained in detail in the following subsections.

3.2 *TinyOS*

TinyOS is a popular sensor node operating system[11]. It features a component-based architecture which enables rapid innovation and implementation while minimizing code size as required by the severe memory constraints inherent in sensor networks.

TinyOS's component-based and event-driven execution model enables fine-grained power management and yet allows some scheduling flexibility that is necessary due to the unpredictable nature of wireless communication and physical world interfaces. However, *TinyOS* is unable to support multimodal tasking well, and moreover, it does not consider real-time scheduling of these tasks and thus does not fit for real-time sensor network systems.

3.3 *SOS*

SOS is an operating system for *Mote*-class wireless sensor networks[10]. It uses a common kernel that implements messaging, dynamic memory, module loading

Table 1. Comparison of sensor node operating systems

	Operating Systems			
Features	TinyOS	SOS	MANTIS	Nano-Qplus
Low power mode support	Y	Y	N	Y
Multimodal sensing/tasking	N	Y	Y	Y
Dynamic reprogramming	Y	Y	N	N
Priority-based Scheduling	N	N	Y	Y
Real-time guarantee	N	N	Y	Y
Execution model	Component based	Module based	Thread based	Thread based

and unloading, and other services. *SOS* uses dynamically loaded software modules to create a system supporting dynamic addition, modification, and removal of network services. One of *SOS*'s primary motivation and goal is to achieve dynamic reprogramming. In the domain of wireless sensor networks, reprogramming is necessary to modify the software on individual nodes after the networks have been deployed. This function provides the ability to update some of the software modules in individual nodes, add new modules to nodes after deployment.

In *SOS*, modules are not processes, they are scheduled cooperatively and they are independent of each other. Therefore, *SOS* does not have a global real-time scheduler and thus is unable to guarantee the real-time schedule of modules.

3.4 *MANTIS*

MANTIS provides a thread-based embedded operating system for wireless sensor networks[1]. *MANTIS* supports preemptive multithreading. It also enables sensor nodes to natively interleave complex tasks with time-sensitive tasks, thereby mitigating the bounded buffer producer-consumer problem. In other words, finely interleave concurrency of multithreading is useful in sensor node systems to prevent one long-lived task from blocking execution of a second time-sensitive task. For example, *TinyOS* does not consider this problem, but *MANTIS* have solved the problem by using the concept of thread.

3.5 *Nano-Qplus*

Nano-Qplus was designed and developed to meet two kinds of demands of wireless sensor networks[4]. The first one is scalability and reconfigurability. Existing sensor network systems designs cannot be easily used to application areas due to their variety of environments and the sensor hardware platforms, etc. In *Nano-Qplus*, various hardware platforms converge into identical system model by the

Nano-HAL hardware abstraction layer, so the programmers can easily make programs for their purposes. The second one is thread-based priority task scheduler. Therefore, similarly to *MANTIS*, *Nano-Qplus* can be used in real-time sensor network systems.

4 Performance Analysis

In this section, we present an evaluation criteria of sensor node operating systems and will evaluate and analyze performance of existing sensor node operating systems. Furthermore, we will discuss some possible improvements for existing sensor node operating systems.

4.1 Experimental Setup

In our experiment, we used *Octacomm's Nano-24* wireless sensor platform[17]. The specification of *Nano-24* sensor platform is given in Table 2.

Table 2. *Nano-24* sensor platform specification

Component	Model	Description
CPU	ATmega128L	low-power 8bit microprocessor 8 Mhz clock
Memory	Flash SRAM EEPROM	128 KB 4 KB 4 KB
RF	CC2420	2.4 Ghz channel Zigbee support 250 Kbps rate
Power	2×AA Batteries	3.0 Volt

Table 2 shows the hardware components and its description of *Nano-24* sensor board. It is similar to *MICAz* sensor board, however, its architectural details are slightly different. *Octacomm* supports *TinyOS* and *Nano-Qplus*' kernel source for *Nano-24* sensor board. But in *SOS* and *MANTIS*, there is no adequate kernel source for this sensor board. For our experiment, we ported *SOS* and *MANTIS* to *Nano-24* sensor board[1].

4.2 Evaluation Criteria

In our experiments, we focused on the efficiency of task scheduler and task-related kernel routines because the efficiency of the task-related kernel routines affect the efficiency of the whole operating system. The selected evaluation criteria for our experiments are presented below.

[1] *Used kernel version of the operating systems; TinyOS: 1.1.11-3, SOS: 05-july, MANTIS: 0.9.1b, Nano-Qplus: 1.6.0e.*

- *Task creation latency*
- *Memory allocation latency*

Task creation latency means the timing overhead of the task scheduler, thus it is necessary for our experiments. *Memory allocation latency* occurs whenever a task is created, or if a task requires more memory space. It is basically the timing overhead of task management and scheduling component in operating system.

4.3 Experimental Results and Evaluation

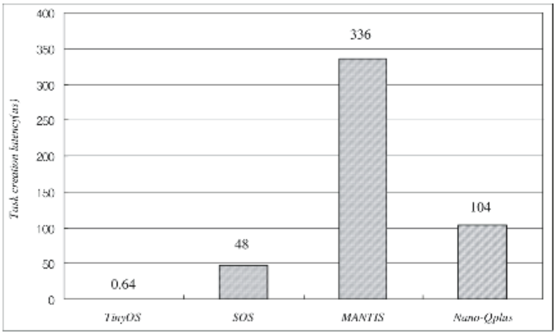


Fig. 2. Task creation latency of operating systems

Figure 2 shows the task creation latency on existing operating systems[2]. From this figure, the results of the task creation latency are significantly different from one another because the characteristics of these operating systems are slightly different. First of all, *TinyOS*' latency is much smaller than the others because *TinyOS*' task creation simply means assigning function pointer of a task to a ready queue. It does not need memory to be allocated or copied because *TinyOS*' scheduler is *FIFO*(non-preemptive). However, *SOS*, *MANTIS* and *Nano-Qplus* operating systems requires memory allocation of task control block. *SOS* is originally designed as a module-based system, and thus needs to allocate the module's address spaces. Meanwhile, *MANTIS* and *Nano-Qplus* are designed in thread-based priority scheduling, and thus the costs of stack allocation and priority queue management are needed.

Figure 3 shows the memory allocation latency on existing operating systems[3]. *TinyOS* does not have memory allocation function and thus memory allocation does not occur. Therefore, we excluded *TinyOS* in this experiment. The x-axis is the amount of allocated memory while the y-axis is latency. In Fig. 3, *SOS* shows best performance and the next is *Nano-Qplus*. In the results, *MANTIS*' memory

[2] *Task creation function of operating systems; TinyOS: Tos_post(), SOS: ker_register_task(), MANTIS: mos_thread_new(), Nano-Qplus: pthread_create().*

[3] *Memory allocation function of operating systems; TinyOS: none, SOS: ker_malloc(), MANTIS: mos_mem_alloc(), Nano-Qplus: _pthread_malloc().*

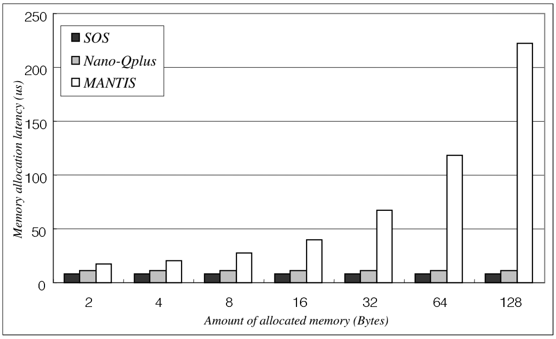


Fig. 3. Memory allocation latency of operating systems

allocation latency increases the amount of allocated memory. These results are significantly different. Thus, we analyzed the source codes of memory allocation function to find the cause of the difference.

SOS uses paging systems of various sizes, and pages consists of 32 × 16bytes, 16 × 32bytes, and 4 × 128bytes of memory space. Some of these are allocated when tasks demand memory allocation. Therefore, memory management of *SOS* has a problem of internal fragmentation. In contrast, *MANTIS* and *Nano-Qplus* uses non-paging system. They uses a first-fit allocation policy which entail searching time cost, and moreover they have a problem of external fragmentation. In addition, *MANTIS* performs memory initialization when allocation is completed, hence increasing memory allocation according to the amount of allocated memory.

4.4 Discussions on Possible Improvements

According to the results of task creation latency, *TinyOS* demonstrates good performance on task creation but, it does not consider preemptive scheduling. However, *Nano-Qplus* considers preemptive priority-based real-time scheduling, but its task creation needs extremely much more time than *TinyOS*. According to the results of memory allocation latency, *TinyOS* does not need to allocate memory, which different from other operating systems that need to allocate task control block to memory space. There is one more aspect about memory fragmentation problem that need to be considered. Since *Nano-24* sensor nodes only have 4 Kbytes of RAM space, the fragmentation poses as a serious problem. Considering the above, the memory allocation latency is important. In this way, we can get trade-off relations between the characteristics of operating systems and the performances.

Let us assume the existence of two kinds of tasks in the operating systems. One is non real-time tasks while the other is real-time tasks. In this case, we can use a *hybrid* task scheduler. The non real-time tasks can be handled by using the mechanism of *TinyOS*, and the real-time tasks can be managed by the priority-based scheduler like *Nano-Qplus*. This will assure better performance than before. Also the memory management and allocation method can be improved by using hybrid approaches. Paging systems such as *SOS* is used when relatively small amount of internal fragmentation occurs. However if a large amount of internal fragmentation occurs, methods such as *Nano-Qplus* is applied.

5 Conclusions and Future Work

In most cases of wireless sensor networks, energy efficiency of sensor nodes is a very important criteria because the power supplied to sensor nodes is very limited. The extensive research have been conducted in connection with energy efficiency for sensor nodes. In this paper, we presented an overview of existing sensor node operating systems and its various functionalities. In addition, we evaluated performances of these operating systems and thus were able to show the advantages and the disadvantages of existing operating systems. Next, we discussed some possible improvements for increasing efficiency of sensor node operating systems for wireless sensor networks.

We are currently in the process of designing and implementing more efficient operating systems based on existing research. We are convinced that applying the above possible improvements on developing sensor node operating systems will ensure enhanced efficiency of the operating systems for wireless sensor networks.

References

1. Bhatti, S., Carlson, J., Dai, H., Deng, J., Rose, J., Sheth, A., Shucker, B., Gruenwald, C., Torgerson, A., Han, R.: Mantis os: An embedded multithreaded operating system for wireless micro sensor platforms. ACMKluwer Mobile Networks and Applications (MONET) Journal, Special Issue on Wireless Sensor Networks (2005)
2. Shah, R., Rabaey, J.: Energy aware routing for low energy ad hoc sensor networks. In: Proc. IEEE Wireless Communications and Networking Conference(WCNC). (2002)
3. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless microsensor networks. In: Hawaii International Conference on System Sciences (HICSS). (2000)
4. Lee, K., Shin, Y., Choi, H., Park, S.: A design of sensor network system based on scalable and reconfigurable nano-os platform. In: IT-Soc International Conference. (2004)
5. Srivastava, M., Muntz, R., Potkonjak, M.: Smart kindergarten: Sensor-based wireless networks for smart developmental problem-solving environments. In: The 7th Annual International Conference on Mobile Computing and Networking. (2001)
6. Akyildiz, I., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A survey on sensor networks. IEEE Communications Magazine (2002) 102–114
7. Lundquist, J.D., Cayan, D.R., Dettinger, M.D.: Meteorology and hydrology in yosemite national park: A sensor network application. Lecture Note in Computer Science **2634** (2003) 518–528
8. Hirafuji, M., Fukatsu, T., Hu, H., Kiura, T., Laurenson, M., He, D., Yamakawa, A., Imada, A., Ninomiya, S.: Advanced sensor-network with field monitoring servers and metbroker. In: CIGR International Conference. (2004)

9. Chen, A., Jain, N., Pietraszek, T., Rooney, S., Scotton, P.: Scaling real-time telematics applications using programmable middleboxes: A case study in traffic prediction. In: 1st IEEE Consumer Communication and Networking Conference. (2004) 388–393
10. Han, C.C., Kumar, R., Shea, R., Kohler, E., Srivastava, M.B.: A dynamic operating system for sensor nodes. In: MobiSys. (2005) 163–176
11. Levis, P., Madden, S., Gay, D., Polastre, J., Szewczyk, R., Woo, A., Brewer, E., Culler, D.: The emergence of networking abstractions and techniques in tinyos. In: First USENIX/ACM Symposium on Networked Systems Design and Implementation (NSDI 2004). (2004)
12. Levis, P., Culler, D.: Mate: a virtual machine for tiny networked sensors. In: International Conference on Architectural Support for Programming Languages and Operating Systems. (2002) 85–95
13. Martin, F., Mikhak, B., Silverman, B.: Metacricket: A designer's kit for making computational devices. IBM System Journal **39** (2000)
14. Schmid, T., Dubois-Ferriere, H., Vetterli, M.: Sensorscope: Experiences with a wireless building monitoring sensor network. Workshop on Real-World Wireless Sensor Networks (REALWSN'05) (2005)
15. Volgyesi, P., Ledeczi, A.: Component-based development of networked embedded applications. In: 28 th Euromicro Conference (EUROMICRO'02). (2002)
16. Crossbow: http://www.xbow.com/. (2005)
17. Octacomm: http://www.octacomm.net/. (2005)

Wireless Sensor Networks: A Scalable Time Synchronization

Kee-Young Shin, Jin Won Kim, Ilgon Park, and Pyeong Soo Mah

Ubiquitous Computing Middleware Research Team,
Embedded Software Research Division,
Electronics and Telecommunications Research Institute, Daejeon, South Korea
{sky64398, john2004, gon, mah}etri.re.kr

Abstract. This paper presents a novel Chained-RIpple Time Synchronization (CRIT) protocol that is scalable, flexible, and high-precise in Wireless Sensor Networks (WSN). CRIT adopts hierarchical and multi-hop time synchronization architecture with contributing energy-saving effects in WSN. The algorithm works in two phases. In the first phase, a horizontal structure between Missionary Nodes (MN) is established in the network by Piggy-Back Neighbor Time Synchronization (PBNT) algorithm. In the second phase, a vertical structure between a MN and Sensor Nodes (SN) is set up in each sensor group (SG) by Distributed Depth First Search (DDFS) algorithm. By applying these two phases repeatedly, all nodes in WSN efficiently synchronize to each other. For the purpose of performance evaluation, we first study the error sources of CRIT. In addition, we simulate CRIT in terms of synchronization errors of two phases using network simulator.

1 Introduction

Recently, the availability of cheap and small micro-sensors and low power wireless communication enabled the large scaled deployment of sensor nodes in Wireless Sensor Networks (WSN). WSN allows us to address, monitor, and eventually control a wide aspect of real-world problems.

A basic function of WSN is data fusion, i.e. combining data from multiple sensors into high level data. As an example, a vehicle going through a sensor network equipped with acoustic sensors can be detected by different sensor nodes at different moments corresponding to the moments when the vehicle entered the detection range of those nodes. A fusion node receiving the raw information from the sensor nodes can refine it by estimating the speed and the direction of the sensed vehicle. For the application and most other applications, synchronized timestamps between sensor nodes together with position information are essential.

Time synchronization problem has been investigated thoroughly in Internet and wireless LAN. Several technologies such GPS and radio ranging have been used to provide global synchronization in networks. Nevertheless, most of these existing time synchronization methods [1][6] do not consider the limited resource and energy available for long time operation of sensor nodes.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 509 – 518, 2006.

Generally, WSN is composed of a large number of mobile sensor nodes. To operate in such large network densities, we need the time synchronization algorithm to be scalable with a number of mobile nodes being deployed and considerable energy efficiency problem due to the limited battery capacity of sensor nodes. Moreover, existing schemes will need to be extended and combined in new ways in order to provide services that meet the needs of applications with the minimum possible energy expenditures.

In this paper, we proposed a flexible and scalable Chained-Ripple Time Synchronization (CRIT) protocol. Our protocol uses a hierarchical and multi-hop architecture suitable for WSN. The remainder of this paper is organized as follows: Section 2 introduces basic concepts and some assumptions used in this paper. Section 3 elaborates the details of our CRIT. In section 4, we study the error sources of CRIT with numerical analysis. The performance evaluation is discussed in section 5. Finally, section 6 concludes this paper and introduces some suggestions for further improvement of our protocol as future works.

2 Our Network Architecture and Assumptions

We denote the sink node as "Base Station (BS)" that is an original time resource node in our topology. The BS has stronger radio transmission capability than MNs and normal "Sensor Nodes (SN)". Nodes are organized into different interconnected domains, called as "Sensor Group (SG)" formulated by Ripple Phase explained later. There is a "Missionary Node (MN)" that is also a time resource node to synchronize with normal SNs in each SG. A MN is selected by the BS or another MN of neighboring group by PBNT algorithm explained later. When a SN becomes a MN having own SG, it is supposed to have stronger radio transmission power than other normal SNs. MNs can manage the state information of all normal SNs in own SG, such as time information, computing resources, and locations.

In addition, we assume that SNs locates in intersect area of SGs only formulate a communication with a MN. Based on the characteristics of these nodes, we categorized them into two-levels. The level-1 node is the BS and MNs and the level-2 nodes are normal SNs.

3 Chained-Ripple Time Synchronization (CRIT)

By applying a hierarchical and multi-hop architecture in WSN, the proposed algorithm is divided into the two phases: Horizontal Missionary Node Discovery Phase (Chained Phase) and Vertical Sensor Node Synchronization Phase (Ripple Phase).

3.1 Horizontal Missionary Node Discovery Phase (Chained Phase)

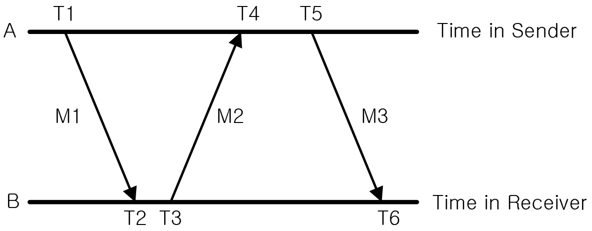
When the BS has been setup for time synchronization in WSN, Chained Phase starts. The BS initially broadcasts MN-REQUEST packet (MRP) through the network for assigning a MN (Step 1). When a normal SN that wants to be a MN receives the MRP, it sends acknowledgement (ACK) packet with own piggybacked clock information to the BS (Step 2). The BS receives this ACK packet and resends MISSIONARY-ASSIGN packet (MAP) with own piggybacked clock information to the SN (Step 3).

Eventually, the SN receiving this MAP becomes a MN in WSN and adjusts own clock information by Piggy-Back Neighbor Time Synchronization (PBNT) algorithm (Step 4). And then, the MN constructs a SG with neighbor SNs by Ripple Phase mechanism (Step 5) and broadcasts a MRP for assigning other MNs same as the BS's operations (Step 6).

3.2 Piggy-Back Neighbor Time Synchronization (PBNT)

The following section describes a basic scheme of Piggy-Back Neighbor Time Synchronization (PBNT) algorithm between the BS and MNs or MNs. PBNT algorithm uses the classical approaches of sender-receiver synchronization [7] with some modifications.

In Figure 1, there are two nodes which are called A and B. A is the BS or a MN in level 1 and B is a normal SN in level 2 that will become a MN having own SG by taking a MAP. T2, T3, and T6 are the time measured according to node B's local clock; T1, T4, and T5 are the time measured according to node A's local clock.



M1 = Broadcast MN-REQUEST packet for assigning Missionary Node
M2 = ACK with node B's clock information
M3 = MISSIONARY-ASSIGN packet with node A's clock information

Fig. 1. PBNT algorithm of CRIT

At T1, node A broadcasts a MRP (M1). Node B receives this MRP at T2. Node B waits for some random time (T3 – T2) before it initiates the two-way message exchange with node A. At time T3, node B sends ACK packet in response to the MRP with own piggybacked clock information (M2) to node A, and node A receives this ACK packet at time T4, where T4 is equal to T3 + o + d. Here, o and d represent the clock offset between the two nodes and propagation delay respectively. At T5, node A sends back a MAP with own piggybacked clock information (M3) to node B. This packet contains a MAP-SET-BIT for assigning a MN and time values of T3, T4, and T5. Eventually, node A receives the packet at T6.

Through the time values of T3, T4, T5, and T6, node B can calculate clock offset and propagation delay as follows:

$$o = \frac{(T4 - T3) - (T6 - T5)}{2} \quad (1)$$

$$d = \frac{(T4 - T3) + (T6 - T5)}{2} \quad (2)$$

The node B can correct its time information and synchronize with node A by referencing results from equation (1) and (2).

3.3 Vertical Sensor Node Synchronization Phase (Ripple Phase)

Ripple Phase indicates the step 5 and 10 in Section 3.1. In this phase, all SNs in each SG efficiently synchronize to each other with a MN by Distributed Depth-first-Search (DDFS) [2] algorithm. At time T6 in Figure 1, the node B, which is a MN built up by a MAP, formulates a DDFS communication link with neighbor SNs and sends its clock information packet. Through this clock information packet, each SN can efficiently tune up own clock information. Here, having a communication and time complexities of O(|N|), where N is the number of node, DDFS algorithm maximized time synchronization accuracy.

Specially, we used a DDFS algorithm different from the previous DDFS algorithms. In previous DDFS algorithm [8], an ACK packet is sent from each notification about sending packet and the sender node holds its information until all notifications are acknowledged. However, in CRIT, no ACKs are used for reducing packet communication overhead and energy consumption, so no time is spent on waiting for them (NO-ACK mechanism). Therefore, packets in this phase are forwarded immediately to the next node. Through these schemes, DDFS algorithm of CRIT can support fast and energy-efficient time synchronization in WSN.

4 Chained-Ripple Time Synchronization (CRIT)

4.1 Sources of Time Synchronization Error

The sources of packet delay are divided as below when it traverses over a wireless link between two sensor nodes. We designate the node which initiates the packet exchange as the sender and the node which responds to this message as the receiver. Although a similar decomposition of error sources has also been presented in [3], we analyze in detail the various delay components from a systems perspective. In this discussion, we will borrow terms from a typical layered architecture used in traditional computer networks.

- ***Send time:*** The time spent at the sender to constructing the packet. This time includes the delay generated by the packet to reach the MAC layer from the application layer, kernel protocol processing, and variable delays introduced by the operating system, e.g. context switches and system call overhead.
- ***Access time:*** The Delay time incurred waiting for access to the transmit channel. After reaching the MAC layer, the packet waits until it can access the channel. This delay time is specific to wireless networks resulting from the property of common medium for packet transmission.

- ***Propagation time:*** The time needed for the packet to transmit from senders to receivers once it has left the sender. When the sender and receiver share access to the same physical media, this time is very small as it is simply the physical propagation time of the message through the media. If not, propagation time dominates the delay in wide-area networks, where it includes the queuing and switching delay at each node as the packet transits through the network.
- ***Accept time:*** The time taken in receiving the bits and passing them to the MAC layer. The variation in reception delay would even be smaller if the sensor node uses a hardware-based RF transceiver [4].
- ***Receive time:*** The time spent by receiver that constructs the bits into a packet and then this packet is passed on the upper layer, application layer where it's decoded. The value of receive time changes due to the variable delays introduced by the operating system.

4.2 Error Analysis of CRIT

In general the hardware clock of node i is a monotonically non-decreasing function of t. In practice, a quartz oscillator is used to incur the real time clock. The oscillator's frequency depends on the ambient conditions, but for relatively extended periods of time (minutes – hours) can be approximated with good accuracy by an oscillator with fixed frequency:

$$t_i(t) = a_i t + o_i \tag{3}$$

Where a_i and o_i are drift and offset of nodes i's clock. In general a_i and o_i will be different for each node and approximately constant for an extended period of time.

From (3) it follows that a_i and o_i are linearly related:

$$t_1(t) = a_{AB} t + o_{AB} \tag{4}$$

The parameters a_{AB} and o_{AB} represent the relative clock drift and offset between the two node's clocks. If the two node's clocks are perfectly synchronized, the relative drift is equal to one and the relative offset is equal to zero.

We can expand this equation (4) to our CRIT by showing the expression of relation between node A and B in Figure 1. After node A and B are synchronized by PBNT algorithm, the two nodes' relation expression can be derived as follows:

$$t_1(t) = a_{AB} t + o_{AB} + d_{AB} \tag{5}$$

The parameters d_{AB} represent the relative propagation delay between two nodes after synchronized to each other.

To be better analysis, we introduce the concept of real time i.e. the time measured by an ideal clock as shown in Figure 2. We represent the times measured by local node clocks, such as T3, in real time by using lowercase letters. Therefore, t3 stands

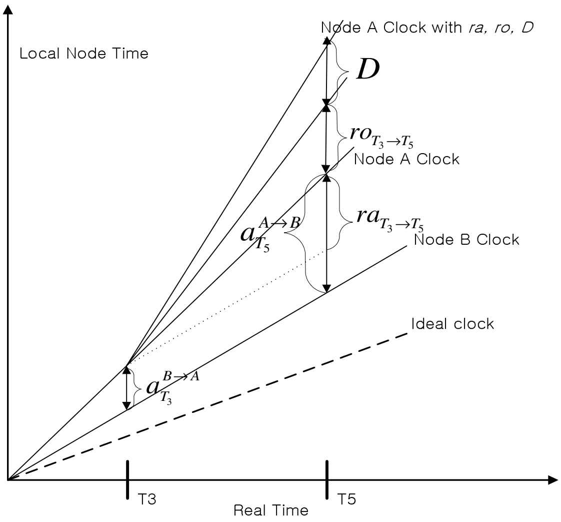


Fig. 2. The time error sources among the local node time

for the real time (measured by ideal clock) equivalent of T3 (measured by node B clock). We apply this mechanism to PBNT algorithm in Figure 1. Node B sends a packet with own clock information at T3 and node A receives it at T4. Note that T3 and T4 are times measured by node's local clock of A and B respectively. The following set of equations can be easily derived:

$$t_4 = 1 \cdot t_3 + 0 + (S_B + AS_B + P_{B \to A} + AC_A + R_A) \tag{6}$$

$$= t_3 + (S_B + AS_B + P_{B \to A} + AC_A + R_A) \tag{7}$$

$$T_4 = a_{T_3}^{B \to A} T_3 + O_{T_3}^{B \to A} + (S_B + AS_B + P_{B \to A} + AC_A + R_A) \tag{8}$$

Here, S_B and AS_B represent the time taken to send packet (send time + access time) at node B. $P_{B \to A}$ refer to the propagation time between node B and A. AC_A and R_A represent the time taken to receive packet (accept time + receive time) at node A. Therefore, equation (8) presents the relationship with local clock mechanism between node A and B after node A sends an ACK packet with own clock information at time T3 and node B receives the packet.

In addition, if we apply this relation to node B at time T6, the next equation is drawn:

$$T_6 = a_{T_5}^{A \to B} T_5 + O_{T_5}^{A \to B} + (S_A + AS_A + P_{A \to B} + AC_B + R_B) \tag{9}$$

After subtracting the equation (9) from (8), we can obtain the following equations easily:

$$\begin{aligned}
T_4 - T_6 &= \{a_{T_3}^{B\to A} T_3 + O_{T_3}^{B\to A} + (S_B + AS_B + P_{B\to A} + AC_A + R_A)\} \\
&\quad - \left\{a_{T_5}^{A\to B} T_5 + O_{T_5}^{A\to B} + (S_A + AS_A + P_{A\to B} + AC_B + R_B)\right\} \\
&= (a_{T_3}^{B\to A} T_3 - a_{T_5}^{A\to B} T_5) + (O_{T_3}^{B\to A} - O_{T_5}^{A\to B}) \\
&\quad + (S_B - S_A) + (AS_B - AS_A) + (P_{B\to A} - P_{A\to B}) \\
&\quad + (AC_A - AC_B) + (R_A - R_B)
\end{aligned} \tag{10}$$

Here, for simplifying the equation (10), we present the following equation like this and substitute it into the upper equation:

$$\begin{aligned}
a_{T_5}^{A\to B} &= a_{T_3}^{B\to A} + ra_{T_3\to T_5} \\
O_{T_5}^{A\to B} &= O_{T_3}^{B\to A} + ro_{T_3\to T_5} \\
S_B - S_A &= S_D \\
AS_B - AS_A &= AS_D \\
P_{B\to A} - P_{A\to B} &= P_D \\
AC_A - AC_B &= AC_D \\
R_A - R_B &= R_D \\
S_D + AS_D + P_D + AC_D + R_D &= D
\end{aligned} \tag{11}$$

Consequently, we can obtain the following equation easily:

$$T_4 - T_6 = a_{T_3}^{B\to A} T_3 - (a_{T_3}^{B\to A} + ra_{T_3\to T_5}) T_5 + O_{T_3}^{B\to A} - (O_{T_3}^{B\to A} + ro_{T_3\to T_5}) + S_U + D \tag{12}$$

Here, $ra_{T_3\to T_5}$ *and* $ro_{T_3\to T_5}$ represent relative clock drift and offset between T_3 and T_5 respectively. D is the total transmission delay between T_3 and T_5. This relation is graphically appeared in Figure 2.

Eventually, we knows from equation (12) that ra, ro, and D are critical elements that impact the synchronization error rate of CRIT. Furthermore, we can obtain the normalized equation of average time synchronization error of Chained Phase in n-hop network as follows:

$$Chained\ Phase\ Error = \frac{1}{n}\sum_{i=1}^{n}\left\{ra_i + ro_i + D_i\right\} \tag{13}$$

Note that, because time synchronization of Ripple Phase using DDFS algorithm is dependent on node's counts in each SG, we regard its time synchronization error rate as $O(n)$, n is the number of nodes. As a result, we can derive the total average synchronization error of the CRIT as follows:

$$CRIT\ Error = \frac{1}{n}\sum_{i=1}^{n}\left\{ra_i + ro_i + D_i\right\} + O(n) \tag{14}$$

Eventually, we can also appear a synchronization error equation of CRIT with maximum and minimum rate as follows:

$$\underline{CRITError} \le \frac{1}{n}\sum_{i=1}^{n}\{ra_i + ro_i + D_i\} + O(n) \le \overline{CRITError} \tag{15}$$

In the next section, we significantly evaluated the time synchronization error of Chained Phase with the upper-bound and lower-bound. And then, we obtained the time synchronization error of Ripple Phase for reflecting results of equation (15).

5 Performance Evaluations

In order to evaluate the performance of CRIT, we simulated it using an extended version of discrete event network simulator NS-2, implemented by the Monarch project [5]. A simple random topology is used, where the BS is preset in the center of our network grid. Five nodes for MNs are automatically selected by Chained Phase when CRIT starts from the BS. According to the number of MNs, five SGs are formulated and each group has one hundred of normal SNs. We placed SNs on a predefined square geographical coverage area with dimension 3000x3000 meters in a uniformly random fashion. Moreover, in order to reflect Ripple Phase mechanism of our CRIT, we implemented the DDFS algorithm in NS-2 and used the IEEE 802.11b MAC protocol with some modifications. The data rate periodically sent from the BS is set to 2Mb/s and we simulated for 3000s.

5.1 Evaluation Metrics

In order to evaluate the performance of Chained Phase and Ripple Phase of CRIT, the following metrics are investigated.

1) Synchronization error of Chained Phase: The time synchronization error including upper-bound and lower-bound of Chained Phase based on PBNT algorithm according to hop distances.

2) Synchronization error of Ripple Phase: The time synchronization error of Ripple Phase using DDFS algorithm according to the number of SNs in each SG.

5.2 Simulation Results

Fig. 3 shows the time synchronization error of Chained Phase with minimum (MIN), average (AVG), and maximum (MAX) values in accordance with MN's hop distances from BS. We evaluate the performance with 5-hop distances by using 1 Base Station and 5 Missionary Nodes. Here, each MIN, AVG, and MAX respectively appears an average time synchronization errors obtained by 1000 simulation runs in each hop distance. As expected, we find that the synchronization errors experienced by PBNT algorithm increase gradually according to increasing hop distances. However, the increasing amount is very slight because the PBNT algorithm used the piggybacked time information mechanism for reducing communication overhead. Therefore, the mechanism of Chained Phase achieves a fine-tuned time synchronization betweens the BS and MNs or MNs.

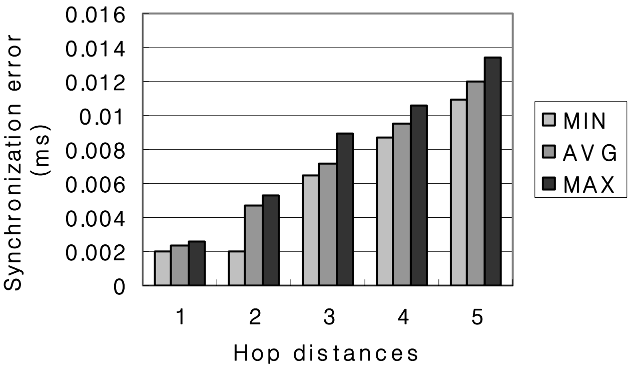


Fig. 3. Synchronization error of Chained Phase

In Figure 4, we reported an average time synchronization error of Ripple Phase with respect to the number of nodes. Every data point represents an average of 1000 simulation runs. Since DDFS algorithm has the communication and time complexity with $O(n)$, where n is the number of node, it is certain that synchronization error increases according to the number of nodes. However, because the increasing time synchronization error rate of this phase is very low by using NO-ACK mechanism for reducing the communication overhead, it is considerable less sensitive in dense WSN. Hence, we can easily notice that Ripple Phase mechanism of CRIT supports accurate time synchronization between a MN and a variety of SNs in WSN.

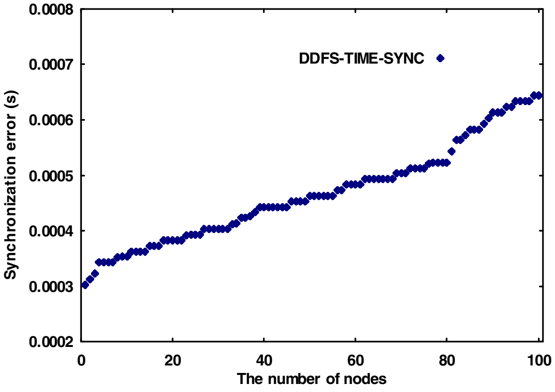


Fig. 4. Synchronization error of Ripple Phase

6 Conclusions

In this paper, we introduced a scalable, flexible, and high-precise time synchronization mechanism with supporting the energy-saving effects in WSN, which is CRIT (Chained-RIpple Time Synchronization). CRIT contributes in the accurate hierarchical and multi-hop time synchronization with low error-rate in WSN. The simulation results satisfied these goals with respect to synchronization errors of Chained Phase and Ripple Phase. Furthermore, because of these characteristics, our CRIT can be efficiently expanded to real WSN products and used for time critical applications in real world. In the future, we plan to evaluate a performance of CRIT's clock offset according to elapsed time.

References

1. L. Lamport, "Time, clocks, and the ordering of events in a distributed system," *Communications of the ACM*, vol. 21, no. 7, pp. 558-565, 1978.
2. Cidon, Yet another distributed depth-first-search-algorithm, Inform. Process. Lett, 26, (6), pp. 301-305, Jan 1988.
3. Jeremy Elson, Lewis Girod and Deborah Estrin, "Fine-Grained Network Time Synchronization using Reference Broadcasts," *In the proceedings of the fifth symposium on Operating System Design and Implementation (OSDI 2002)*, December 2002.
4. Chipcon CC1000 Radio Datasheet,

   http://www.chipcon.com/files/CC1000_Data_Sheet_2_1.pdf
5. The NS-2 simulator, http://www.isi.edu/nsnam/ns
6. Duda, G. Harrus, Y. Haddad, and G. Bernard, "Estimating global time in distributed systems," in *Proc. Of the 7th IEEE International Conference on Distributed Computing Systems (ICDCS'87)*, Berlin, Germany, Sept. 1987.
7. D. L. Mills, "Internet time synchronization: The Network Time Protocol" In Z. Yang andT.A. Marsland, editors, *Global States and Time in Distributed Systems.* IEEE Computer Society Press, 1944.
8. Awerbuch, A new distributed depth-first-search algorithm, Inform. Process. Lett. 20 (3), pp147-150, 1985.

A New Cluster Head Selection Scheme for Long Lifetime of Wireless Sensor Networks[*]

Hyung Su Lee[1,2], Kyung Tae Kim[1], and Hee Yong Youn[1,**]

[1] School of Information and Communications Engineering,
Sungkyunkwan University, Korea
{hsleel, harisu}@skku.edu, youn@ece.skku.ac.kr
[2] Korea Electronics Technology Institute, Korea
hslee@keti.re.kr

Abstract. Wireless sensor network (WSN) consisting of a large number of small sensors can be an effective tool for gathering information in a variety of environments. Since sensor nodes operate on batteries, energy efficiency is a key issue in designing the network. In the existing clustering-based routing protocols for the WSN including the LEACH scheme, the cluster-heads are usually selected at random, which may cause unbalanced energy consumption and thus short network lifetime. In this paper we identify that position of cluster-head is an important factor with respect to the network lifetime. Based on this observation, we propose a scheme which selects the cluster-heads not randomly but considering the relative position and residual energy of the alive nodes in the network. Computer simulation reveals that the proposed scheme extends the lifetime of the network employing the LEACH scheme for about 50%.

Keywords: Cluster-head, energy-efficiency, network lifetime, wireless sensor networks.

1 Introduction

Wireless sensor network (WSN) consists of a large number of tiny sensor nodes forming a distributed wireless ad hoc sensing network. The sensors collect quite detailed information on the physical environment. It has been rapidly developed and widely used in both the military and civilian applications such as target tracking, surveillance, and security management [1, 2]. A sensor node has four basic components; a sensing unit, a processing unit, a radio unit, and a power unit. All the units fit into a matchbox-sized module [3].

Since a sensor node has limited sensing and computational capability and is able to communicate only within short distances, the sensor network operates with the corporative effort of hundreds or thousands sensor nodes. Once the sensor nodes are deployed, they automatically establish the route and then sense the surroundings, process the sensed data, and transmit the result to the base station (BS). As a sensor

[*] This research was supported by the Ubiquitous Autonomic Computing and Network Project, 21st Century Frontier R&D Program in Korea and the Brain Korea 21 Project in 2005.

[**] Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 519 – 528, 2006.

node has to operate for a relatively long duration on a tiny battery, energy efficiency is a main concern. One of the most restrictive factors on the lifetime of wireless sensor networks is the limited energy resource of the deployed sensor nodes. Because sensor nodes carry limited and generally irreplaceable power source, the protocols designed for the WSN must take the issue of energy saving into consideration. Clustering-based routing protocol is a popular routing protocol proposed for the WSN to minimize the consumption of the energy of the sensors. Here, at regular intervals, a set of cluster-heads (CHs) are selected and the other sensor nodes (member nodes) are clustered around the cluster-heads according to a specific clustering algorithm. In the WSN the sensed data sent from adjacent member nodes are usually similar and therefore aggregated by the CHs to reduce data redundancy.

There exist several clustering-based routing protocols proposed for minimizing the energy consumption [5-7]. In the existing clustering-based routing protocols for the WSN including the LEACH (Low-Energy Adaptive Clustering Hierarchy) scheme [5], the CHs are usually selected at random, which may cause unbalanced energy consumption and thus short network lifetime. In this paper we identify that position of CH is an important factor with respect to the network lifetime. Based on this observation, we propose a scheme which selects the CHs not randomly but considering the relative position and residual energy of the alive nodes in the network. Computer simulation reveals that the proposed scheme extends the lifetime of the network employing the LEACH scheme for about 50%.

The remainder of the paper is organized as follows. Section 2 presents a review of the related work reported in the literature. Section 3 introduces the proposed scheme. Section 4 evaluates the performance of the proposed scheme by computer simulation, and compares it with direct communication and LEACH. Finally, Section 5 concludes the paper and outlines future research directions.

2 The Related Works

2.1 The Routing Protocols

The existing protocols proposed for data gathering in the WSN can be classified into hierarchical and non-hierarchical protocol [4] by the way how to organize the sensor nodes. The non-hierarchical protocols include Directed Diffusion [8] and Gossiping [9], while the hierarchical protocols include LEACH [5], PEACH (Proxy-Enabled Adaptive Clustering Hierarchy) [6] and EDACH (Energy-Driven Adaptive Clustering Hierarchy) [7]. Furthermore, the communication pattern adopted for the WSNs takes one of the two general forms; time-driven (periodical) transmission [5-7] and event-driven transmission [8,9].

LEACH is a clustering-based protocol that applies randomized rotation of the cluster-heads to evenly distribute the energy load among the sensor nodes in the network. The CHs of the LEACH scheme collect data from the distributed micro sensors and transmit them to the base station. It adopts the following clustering-model. Some of the nodes elect themselves as cluster-heads in each round of communication. After the cluster-heads are decided, each cluster-head broadcasts an advertisement message. The sensor nodes listen to the advertisements and join the closest cluster-head. After the clusters are formed, the cluster-heads collect sensor data from the member nodes and transfer the aggregated data to the base station. Figure 1 shows the hierarchical structure of the cluster-based scheme. Note here that the clusters are not necessarily the same size and shape. The recent researches on the routing with the hierarchical structure such as PEACH and EDACH employ a similar approach as LEACH.

PEACH [6] is a protocol that improves LEACH in terms of network lifetime. This is achieved by selecting a proxy node which can assume the role of the current cluster-head of weak power during one round of communication. PEACH is based on the consensus of healthy nodes for the detection and manipulation of failure in any cluster-head. It allows considerable improvement in the network lifetime by reducing the overhead of re-clustering.

EDACH [7] employs a similar approach as the PEACH scheme. It, however, further improves the performance of PEACH by varying the number clusters according to the distance from the base station such as more clusters in the region which is far from the base station.

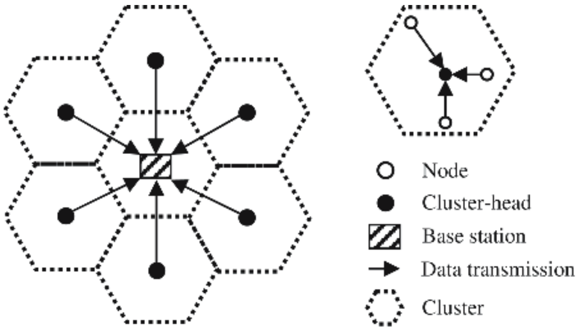


Fig. 1. The structure of hierarchical protocol for wireless sensor networks

2.2 The Energy Issue

In LEACH, the representative protocol of cluster-based sensor network, the cluster-heads are selected randomly and all nodes are given equal opportunity to be the cluster-head. The cluster-heads are selected without considering the position and current energy level, and this may cause unbalanced energy consumption and shortened network lifetime in the long run.

If cluster-headers are selected randomly, they can even locate at the boundary of the network. Since the cluster-heads usually waste more energy than other nodes, this inappropriately positioned cluster-heads will deteriorate the energy efficiency of entire network. With the random selection approach for the cluster-heads, the clusters may be unbalanced in terms of the number of nodes and position of the cluster-heads.

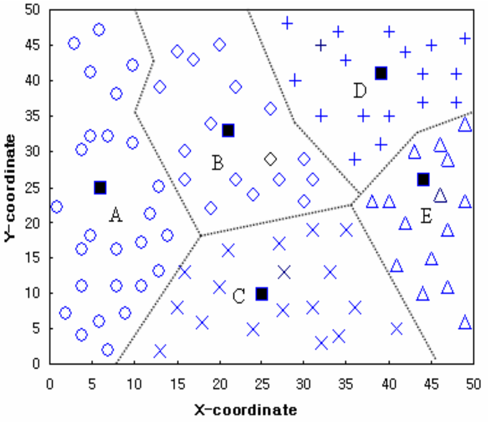


Fig. 2. The sensor network consisting of five clusters

In Figure 2, for example, 100 sensor nodes exist in an area of 50×50 units. Here Cluster-A, B, C, D, E contains 27, 18, 20, 19, 16 nodes, respectively. The black squares are cluster-heads selected randomly. Note that, for the cluster having more member nodes, the cluster-head would have more traffic load and correspondingly waste more energy than the cluster-head of fewer member nodes. Figure 3 shows the energy consumption at each cluster-head and the entire cluster in one round communication when the BS is located at (25, 25).

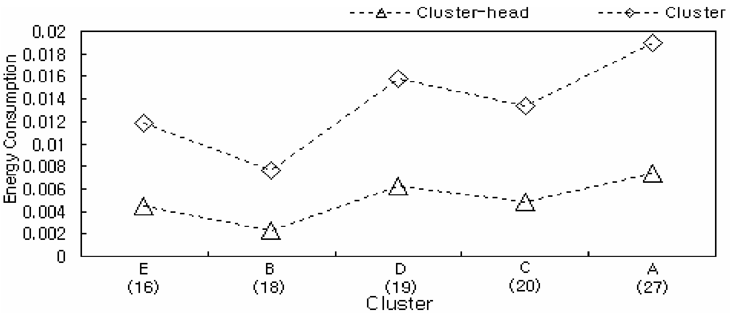


Fig. 3. The energy consumption of cluster-head and entire cluster

Notice from Figure 3 that, even though it is not strictly linear, energy consumption of the cluster-head and entire cluster increases as the number of member nodes increases. Also, energy consumption of Cluster-E of 16 nodes is higher than that of Cluster-B of 18 nodes, which has more number of nodes. This is because the cluster-head of Cluster-E is not located at the center as in Cluster-B. The same situation is observed between Cluster-D and Cluster-C. From this example we can see that energy efficiency of hierarchical routing protocol for the WSN significantly depends on the effectiveness of the selection approach of the cluster-head. This motivates the proposed scheme presented next.

3 The Proposed Scheme

In this section we present the proposed scheme for the selection of cluster-heads in the hierarchical sensor network. A cycle of data gathering, called a round, consists of three phases; (i) cluster-head selection, (ii) cluster formation and schedule creation, and (iii) data collection and transmission. The operations of the sensor nodes in the second and third phase are identical to those with LEACH. We first introduce the main concept of the proposed scheme.

3.1 The Main Concept

In designing the routing protocol for a WSN, the primary goal is long network lifetime (in other words the number of alive nodes at a time moment). In order to achieve the goal, energy consumption of the nodes needs to be well balanced. As we noticed in the previous section, energy consumption of the sensor nodes is significantly affected by the size of the cluster and position of the cluster-head. Another aspect is that cluster-head spends much more energy than other nodes. Therefore, we propose to select the cluster-heads not randomly but deterministically according to the two factors; residual energy and relative location.

First of all, it is our intuition that the node with large residual energy should be selected as a cluster-head. Also, the node having more neighbor nodes than others is preferred as a cluster-head. Note that having a large number of neighbor nodes implies that the node locates at the center position of the area where many nodes are dispersed. It would be beneficial to select such centered nodes as cluster-heads with respect to the overall energy efficiency, compared to the selection of the nodes locating at relatively isolated positions. One question here is, though, which factor between the residual energy and relative location has more impact on the network lifetime. We answer this question by computer simulation in the next section of performance evaluation. We next present the operation of the proposed scheme decided based on the motivation shown above.

3.2 The Operations of One Round Communication

- **Cluster-head Selection Phase**

In every round each node first needs to decide whether it will be a cluster-head in that round. The decision is made based on the threshold, *T*, which is set using the two factors mentioned above; residual energy and relative location. If a randomly generated number between 0 and 1 is smaller than *T*, the node elects itself as a cluster-head. The threshold is given by

$$T = (\alpha \frac{E_r}{E_i} + (1 - \alpha)\frac{n_a}{n_i})p \qquad (1)$$

Here E_r is the residual energy, E_i is the initial energy at deployment time, n_a is the number of alive neighbor nodes, n_i is the maximum number of neighbor nodes among the nodes when first deployed, p is the portion of cluster-heads among the nodes at the beginning, and $\alpha\,(0 \leq \alpha \leq 1)$ is a weighting factor. The first and second additive term represents the energy and location factor, respectively, while α is used for assigning a weight to each of them. Note that the division in each of the term is for the normalization so that their values become between 0 and 1 since $E_r \leq E_i$ and $n_a \leq n_i$. As a result, the sum of the two terms also becomes between 0 and 1. Multiplying ($p \leq 1$) finally makes only some portion of the nodes become the cluster-heads. Also, notice that E_r and n_a decrease as time goes by while E_i and n_i are constant. Therefore, the threshold value also decreases as the rounds proceed, and consequently the probability of each node to become a cluster-head decreases. This is another distinct feature of the proposed scheme compared to LEACH where the probability does not change. It is our conjecture that a smaller number of cluster-heads are required if there exist a smaller number of alive nodes in the network. The validity of this conjecture is confirmed in the next section by computer simulation.

The node that has elected itself as a cluster-head for the current round broadcasts an advertisement message to the rest of the nodes. For the cluster-head advertisement, the cluster-heads use the CSMA MAC protocol. The non-cluster-head nodes keep their receivers on during this phase of set-up to hear the advertisements of all the cluster-head nodes. After this phase is complete, each non-cluster-head node decides the cluster to which it will belong for this round based on the strength of the received advertisement signal.

- **Cluster Formation and Schedule Creation Phase**

After the advertisement, every cluster member node recognizes the source of the token as its cluster-head and broadcasts the topology reply packet back to the cluster-head using the CSMA MAC protocol. During this phase, all cluster-head nodes must keep their receivers on. The cluster-head receives the messages from the nodes that would like to be included in the cluster. When the cluster-heads receive the reply packets, they set up a schedule for the nodes in their cluster. Based on the number of nodes in the cluster, the cluster-head creates a TDMA schedule indicating when each node in the cluster can transmit. This schedule is broadcast back to the nodes in the cluster.

- **Data Collection and Transmission Phase**

After the schedule creation phase, the self-organized data collection and transmission phase starts. Every sensor node collects data and then sends a packet to the cluster-head in its scheduled transmission time. Based on the received signal strength of the cluster header advertisement and the assumption of the symmetrical radio channel, the transmission can use a minimum amount of energy. The radios of other nodes are turned off until their allocated transmission time to save the energy. Each cluster-head keeps its receiver on to collect data from the nodes in its cluster and continuously updates the table listing the energy of the nodes based on the received packets. When the data from all the member nodes are received, the cluster-heads apply data fusion to aggregate the received data into one packet.

4 Performance Evaluation

In this section we evaluate the effectiveness of the proposed scheme along with direct communication and the LEACH scheme through computer simulation. We use the same radio model as in [5] with E_{elec} = 50nJ/bit as the energy being dissipated to run a transmitter or receiver circuitry and E_{amp} = 100pJ/bit/m^2 as the energy dissipation of the transmission amplifier. Transmission (E_{Tx}) and receiving costs (E_{Rx}) are calculated as follows:

$$E_{Tx}(k,d) = E_{elec} \times k + \varepsilon_{amp} \times k \times d^2 \tag{2}$$

$$E_{Rx}(k) = E_{elec} \times k \tag{3}$$

with k as the length of the message in bits, d as the distance between the transmitter and receiver node, and λ as the path-loss exponent ($\lambda \geq 2$). For computer simulation, we let the maximum percentage of cluster-heads be 5% (thus p = 0.05 in eq (1)), with which LEACH has shown the best performance [5]. For the simulation we consider a sensor network of 100 sensor nodes randomly located in a 50×50 region. The base station is located at (25, 25). An example of a randomly generated sensor network is shown in Figure 4. We use two models of initial residual energy of sensor nodes; uniform at 0.5J and random between 0.25J and 0.5J. The size of sensor data is 2000 bits, and the advertisement message is 64-bit long. In the simulation the result of 100,000 runs are averaged.

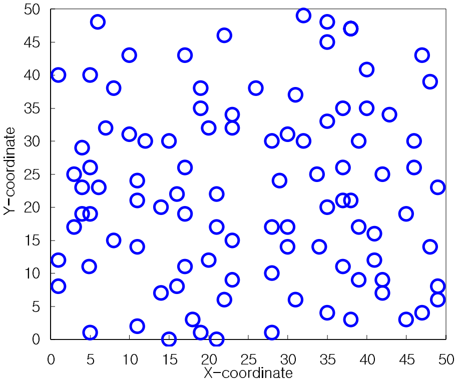


Fig. 4. An example of sensor network with 100 nodes

Table I lists the lifetime of sensor network in terms of the round a node begins to die and the round all the nodes die for the three schemes compared. We considered three sets of α value, 0.2, 0.5, and 0.8. Recall that small value of α implies giving more weight on the position factor than the residual energy factor of a node. Notice from the table that the proposed scheme consistently outperforms the others. Especially, with α = 0.2, the proposed scheme improves the lifetime of LEACH for about 50%. Note that small value of α allows better performance, which means that position of cluster-head is a more important factor than residual energy.

Table 1. The network lifetimes of the compared schemes

Energy (J/node)	Protocol	The round a node begins to die	The round the last node dies
0.25	Direct	78	137
	LEACH	253	562
	Proposed Scheme (α =0.2)	367	634
	Proposed Scheme (α =0.5)	264	651
	Proposed Scheme (α =0.8)	227	608
0.5	Direct	157	282
	LEACH	614	1215
	Proposed Scheme (α =0.2)	966	1264
	Proposed Scheme (α =0.5)	820	1302
	Proposed Scheme (α =0.8)	489	1251

Figure 5 shows the number of alive nodes as the round proceeds for the three schemes. The improvement offered by the proposed scheme over Direct Communication and LEACH can be clearly seen from the figure. Here, each sensor node has initial energy of 0.25J/node in the areas of 50m×50m. In LEACH, every sensor has the same chance to become a cluster-head. A sensor node with insufficient residual energy occasionally becomes a cluster-head, even if there is a sensor node with rich battery power nearby. It exhausts its energy, stops operating, and disrupts gathering of sensor data in its cluster. Also, data transmission to the base station is not possible. On the other hand, the cluster-heads are selected considering the residual energy and relative location in the proposed scheme. As a result, energy consumption can be well distributed among the sensor nodes, and the lifetime of the sensor network can be prolonged. In addition to reducing energy dissipation, the proposed protocol successfully distributes energy-usage among the nodes in the network.

Another important aspect of the proposed protocol is illustrated in Figure 6, which shows the locations of live (circle) and dead (dot) sensor nodes with LEACH and the proposed protocol, respectively, after 540 rounds. Here, each node is equipped with an energy source whose total amount of energy accounts for 0.25J at the beginning of the simulation. Observe that, in addition to a lot more live nodes of 80 than LEACH of 24 nodes, the proposed scheme allows well distributed live nodes. Avoiding any dead spot is another important property of the proposed scheme in addition to extended lifetime.

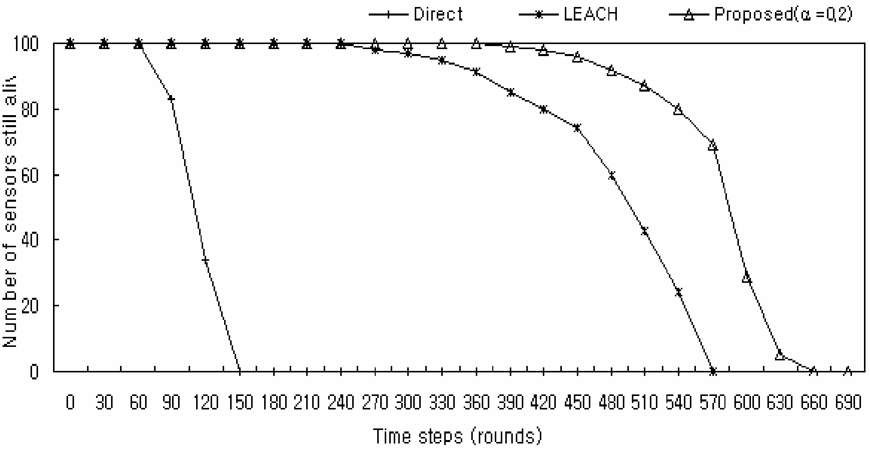
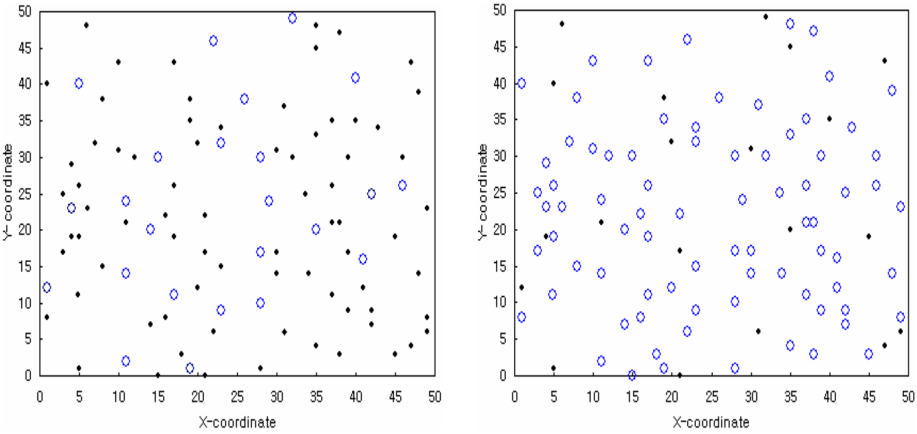


Fig. 5. Comparison of the number of live sensors as the round proceeds



(a) LEACH

(b) Proposed Scheme ($\alpha = 0.2$)

Fig. 6. The distribution of live (circle) and dead (dot) nodes after 540 rounds

5 Conclusion and Future Work

In this paper, for solving the problem of unbalance in the energy consumption of the sensor nodes, we have proposed a new routing scheme for clustering-based sensor network. The scheme selects the cluster-heads according to the residual energy and the number of alive neighbor nodes. Computer simulation shows that position is a more important factor than residual energy of the nodes. It also identifies that the proposed approach extends the lifetime of the sensor network for about 50% compared to the existing schemes randomly selecting the cluster-heads. The proposed approach will be more important when the wireless sensor network is deployed in large area and the base station is far from the network.

The future work will focus on the comparison of the proposed approach with other approaches such as simulated annealing and taboos search. A formal methodology will also be developed in order to determine the factors important for cluster-head selection in a more systematic way and allow optimal results for the given conditions. The current model is based on the one-hop cluster performances. It will be extended for multi-hop clusters. The proposed approach will also be combined with other approaches which improve the network lifetime.

References

[1] L. Zhong, R. Shah, C. Guo, J. Rabaey.: An ultra low power and distributed access protocol for broadband wireless sensor networks. IEEE Broadband Wireless Summit, Las Vegas, May 2001.

[2] K. Sohrabi, J. Gao, V. Ailawadhi, and G. J. Pottie.: Protocols for self-organization of a wireless sensor network: IEEEPersonal Commun., 7(5):16-27, Oct. 2000.

[3] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci.: Wireless sensor networks: a survey: Computer Networks, pp. 393-422, 38 (4) (2002)

[4] Chuan-Ming Liu, Chuan-Hsiu Lee.: Power efficient communication protocols for data gathering on mobile sensor networks: Vehicular Technology Conference, VTC2004-Fall. 2004 IEEE 60th Volume 7 pp. 4635 - 4639, 26-29 Sept, 2004.

[5] W.R.Heinzelman, A.Chandrakasan, and H. Balakrishnan.: Energy-Efficient Communication Protocol for Wireless Micro-sensor Networks: In Proceedings of the Hawaii International Conference on System Science, Maui, Hawaii, 2000.

[6] K.T. Kim and H.Y. Youn.: PEACH: Proxy-Enable Adaptive Clustering Hierarchy for Wireless Sensor network: Proceeding of The 2005 International Conference On Wireless Network, pp. 52-57, June 2005.

[7] K.T. Kim and H. Y. Youn.: Energy-Driven Adaptive Clustering Hierarchy (EDACH) for wireless sensor networks: EUC LNCS3823, pp. 1098-1107, 2005.

[8] C. Intanagonwiwat, R. Govindan, and D. Estrin.: Directed diffusion: a scalable and robust communication paradigm for sensor networks: in Proceedings of ACM International Conference on Mobile Computing and Networking (MobiCom 2000), pp. 56-67, 2000.

[9] S. Hedetniemi and A. Liestman.: A survey of gossiping and broadcasting in communication networks: Networks, vol. 18, no. 4, pp. 319-349, 1988.

Two-Dimensional Priority Scheduling Scheme for Open Real-Time Systems[*]

Pengliu Tan, Hai Jin, and Minghu Zhang

Cluster and Grid Computing Lab.
School of Computer Science and Technology
Huazhong University of Science and Technology, Wuhan, 430074, China
hjin@hust.edu.cn

Abstract. This paper focuses on the scheduling of the tasks with hard, soft and non-real-time timing constraints in open real-time systems. It describes a *Two-Dimensional Priority Scheduling* (TDPS) scheme which not only sets task priority, but also specifies scheduling policy priority. The execution order of a task is determined by both the task priority and its scheduling policy priority. TDPS also supports separating the scheduling mechanism from the scheduling policy. We also enhance TDPS scheme by introducing the CPU utilization bound to each scheduling policy to simplify the schedulability analysis. TDPS scheme can be used to implement different real time systems with different goals (such as hard, soft or hybrid real-time systems) by adjusting the CPU utilization bound of every scheduling policy in runtime. The paper shows through evaluation that TDPS is more open and efficient than the past open real-time scheduling schemes.

1 Introduction

In recent years, the domain of real-time computing has broadened from primarily hard real-time closed embedded systems, such as avionics and automotive applications, to new open environments with other types of performance constraints, such as the Internet and mobile computing systems. In such open environments, independently developed system components and applications share common resources. The real-time systems in open environments are called open real-time systems.

The most important characteristic of open real-time systems is that they are able to process the applications deployed independently. Further, real-time and non-real-time applications are allowed to join and leave the system dynamically. Because of these, open real time systems put a new challenge for task scheduling. The traditional real time scheduling methods, which are dedicated to some special applications, are not suitable for these systems any more. To satisfy the requirements of open real-time systems, the open real-time scheduler must have the following points:

1. Separating the scheduling mechanism from the scheduling policy [1]. The mechanism is in the kernel but the policy is set by a user process. It can permit the user to

[*] This paper is supported by National Science Foundation of China under grant 60125208.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 529 – 538, 2006.

choose the desirable scheduling policy for each application, even for each task in its application.

2. Reconfigurable. The scheduler can be reconfigured, so that the real-time system can meet different requirements.
3. Extensible. A new scheduling policy can easily be added to the scheduler to meet new requirements.
4. Dynamic. The scheduler can accept and schedule the tasks arrival dynamically in runtime.

The motivation of our work is to develop a new efficient scheduling frame for open real-time systems. The rest of the paper is structured as follows. The related work is stated in section 2. Section 3 presents TDPS. In section 4, we evaluate TDPS. Finally, in section 5, some concluding remarks are made.

2 Related Works

Three main scheduling paradigms have been used to schedule real-time tasks, namely, *priority-driven* (PD), *share-driven* (SD), and *time-driven* (TD).

PD scheduling includes two types of algorithms, fixed priority and dynamic priority scheduling. A well-known fixed priority algorithm is RM algorithm [2]. The most popular dynamic priority scheduling algorithms is EDF [2]. Although the PD paradigm can be quite effective, there are situations where such solutions may not be appropriate. For example, in a real-time video-conferencing system, no hard guarantees of real-time performance are required. For such applications, a proportional share resource allocation paradigm, which we call share-driven, may be more desirable.

SD scheduling paradigm is based on the GPS (*General Processor Sharing* [3]) algorithm. Some well-known SD scheduling mechanisms include the *Weighted Fair Queuing* (WFQ) [4], also known as *Packet Generalized Processor Sharing* (PGPS) [3], WF2Q [5], Fair Service Curves [6], etc. Similar ideas have been used in CPU scheduling such as the *Total Bandwidth Server* (TBS) [7], the *Constant Bandwidth Server* (CBS) [8], and the *Constant Utilization Server* (CUS) [9]. One problem of SD is that it may not guarantee a timely completion of hard real-time jobs when the system is overloaded. When the system is overloaded, all channels will receive larger delays proportionally. For systems with steady and well-known input data streams, time-driven (TD) schedulers have been used to provide a very predictable processing power for each data stream [10, 11].

In TD scheduling paradigm, the time instants when each task starts, preempts, resumes and finishes are pre-calculated and enforced by the scheduler. Applications such as small embedded systems, automated process control, sensors can be implemented efficiently using this scheduling paradigm. TD paradigm provides a good predictability and allows off-line schedule optimization. However, TD schedulers are usually less flexible and not suitable for dynamic real-time systems, such as web-based or grid-based real-time systems. Each of the three scheduling paradigms is suitable for special area and special task mode. They can not be used in open real-time systems directly.

A two-level hierarchical scheme for open real-time systems is proposed [9]. The scheme assumes that when the operating system admits a new real-time application into the system, it creates a dedicated constant utilization server to execute the application. At the top level, the operating system allocates processor time to the servers, sets their deadlines, and schedules the servers according to the EDF algorithm. At the low level, the scheduler of the server for each application schedules the tasks within the application according to a priority-driven algorithm chosen for the application. The schedulability of any application can be validated independently of other applications. Non-real-time applications are scheduled in a time-sharing fashion.

An extended two-level hierarchical scheme is proposed [12]. It can accommodate a much broader spectrum of real-time applications. In [13], Kuo and Li follow the open system architecture [9, 12] and replace the underlying OS scheduler with RM scheduler. The two-level hierarchical scheme [9] and the extended schemes [12, 13] have three main shortcomings: (1) the overhead is high, (2) they are not suitable for the applications with parallel threads or processes in parallel or distributed systems (3) they are not suitable for some complex real-time applications consisting of real-time and non-real-time parts. Because of (2), an extended two-level hierarchical scheme in parallel and distributed systems [14] is proposed, but the extension only can improve the concurrency between the applications and can not improve the concurrency between the tasks within one application.

A general real-time scheduling framework is presented in [15, 16]. In the framework, PD, SD and TD paradigms are integrated together. But this is only simple accumulation. Only one paradigm can be adopted when the system runs. The framework does not separate scheduling mechanism from scheduling policy. The system only meets the requirements of a kind of user at one time. So the framework is not suitable for open real-time systems.

RBED [17] meets hard and soft real-time requirements by dynamically controlling the rate that applications consume CPU. RBED model is similar to the *Variable Rate Execution* model (VRE) [18], which also defines rules for changing application rates at any time. The elastic task model [19] similarly defines a method for sets of tasks to simultaneous changing rate, using the novel approach of modeling task utilization as elastic springs. BEBS [20] extends RBED by integrating more robust and efficient best-effort support. BEBS algorithm is similar to IRIS [21], which is based on the CBS [8] but enhances CBS with a fairer slack reclaiming strategy. BEBS differs from these and other aperiodic servers [22, 23] in that BEBS adapts its period and utilization according to the best-effort workload it is presented. But it is a hard thing to adjust periods or rates of tasks to the proper values in runtime in practical systems.

TDPS proposed in this paper is an efficient real-time scheduling framework. It is not only suitable for open real-time systems with one processor but also suitable for parallel and distributed real-time systems.

3 Two-Dimensional Priority Scheduling (TDPS)

All the previous algorithms based on PD almost belong to *One-Dimensional Priority Scheduling* (ODPS). In ODPS, the priority is linear and has only one dimension. Generally, multiple scheduling algorithms based on PD can not coexist in these systems. In ODPS mechanism, the scheduling policy is determined by the system and can not be chosen by users. This deviates from the idea of separating scheduling mechanism from scheduling policy in open real-time systems. Additionally, when there are many real-time tasks in the system, the overhead is very high. There is only one ready queue in the system with ODPS. When the ready queue is very long, the operations of task entering or leaving the ready queue will produce considerable overheads because of looking for the position where the task inserts. Moreover, in the system with ODPS that adopts hybrid PD paradigm with specified-priority scheduling (the priority of a task is specified by the user according to the importance of the task) combined with EDF or RM, every task scheduled by EDF or RM must be assigned a priority number, and when a new task arrives dynamically, the system may reassign the priority numbers for all the tasks scheduled by EDF or RM. These operations will also cause extra overhead.

RTAI [24] is an open source project about real-time Linux. Its scheduling mechanism also belongs to ODPS. But there is a little improvement in its scheduler. It permits EDF and RM to coexist in the system. The priorities of the tasks scheduled by EDF are higher than the priority of any task scheduled by other scheduling policy. Our TDPS just derives from the improvement in RTAI scheduler.

3.1 Architecture of Two-Dimensional Priority Scheduling

TDPS scheme not only sets task priority, but also specifies scheduling policy priority. The priority is not linear any more but has two dimensions. In TDPS, we first set the priorities of scheduling policies. For example, the priority of EDF is set to the highest priority. The priority of RM is set to the second highest priority. The priority of SD is assigned to the second lowest priority. The priority of non-real-time scheduling policy is assigned to the lowest one. The assignment has intrinsic rationality. Second, we assign the priorities to the tasks with the same scheduling policy according to their policy. In practice, actual priority numbers need not be assigned to the tasks. The ways in that they enter their ready queues just can embody the scheduling policies they adopt. In TDPS, the execution order of a task is determined by both its priority and its scheduling policy priority. The ready task with the highest scheduling policy priority and the highest priority is always scheduled first.

But there is a special case that if the ready queue of the scheduling policy with the highest priority is always nonempty, the tasks in the ready queue of the scheduling policy with lower priority will not be scheduled forever, which is called *dead-waiting*. To resolve this problem, we introduce CPU utilization control into TDPS. The main idea is to share CPU among the real-time scheduling policies in the system in some ways and to set the upper bound of the CPU utilization for every real-time scheduling policy. The assignment must be done in such a way that the overall processor utilization of the tasks adopting the same scheduling policy never exceeds a specified maximal value. The sum of all the upper bounds must be less than or equal to one in uniprocessor systems. In the improved scheme, the ready tasks with the highest scheduling policy will be scheduled first. But when the CPU budget for this scheduling policy is used up, the CPU control right will be given to the tasks with the second highest priority policy. The improved scheme can not only guarantee the tasks with high priority policy are processed first but also ensure the tasks with low priority policy can be scheduled. The dead-waiting problem is also removed. Fig.1 shows the architecture of TDPS.

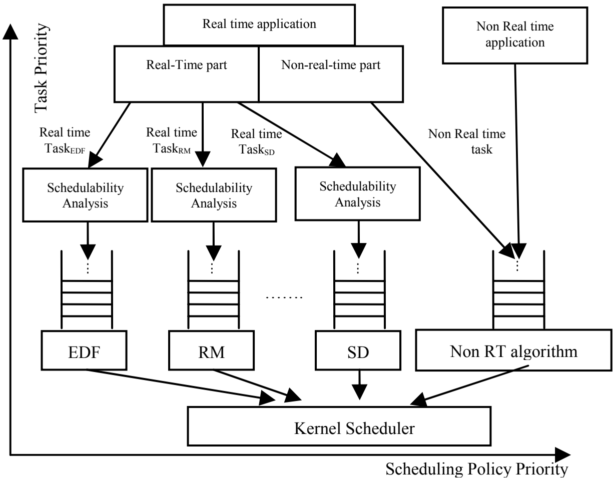


Fig. 1. The architecture of Two-Dimensional Priority Scheduling

TDPS has the following merits. In TDPS, scheduling mechanism is separated from scheduling policy. The system can accept hard, soft and non-real-time applications at the same time. The user can select a scheduling policy for each task in its application. The sporadic hard tasks, periodic hard tasks and multimedia soft tasks can be scheduled by EDF, RM, and SD, respectively. The non-real-time part of an application can be scheduled by non-real-time policy. This satisfies the basic requirement of open real-time systems.

TDPS has high extensibility. New scheduling policy can be added to the scheduler easily.

TDPS has low overhead. The system maintains multiple shorter ready queues. This can reduce the overhead of looking for the location where the new task should be inserted when it becomes ready. The system also can avoid the overhead produced by the dedicated servers in the two-level hierarchical schemes [9, 12, 13].

TDPS can be extended to parallel and distributed real-time systems easily. The tasks in one application can adopt different scheduling policies. Thus, they can be allocated to different processors so that TDPS can not only improve the parallelism between applications but also enhance the concurrency between the tasks within an application.

In TDPS scheme, the schedulability analysis of the tasks with one scheduling policy is independent of the tasks with another policy. The schedulability of a new task only relies on the tasks adopting the same scheduling policy as the new task. Additionally, the scheduler is very flexible and reconfigurable. By adjusting the upper bound of CPU utilization for every policy, the different real-time systems with different goals and QoS can be implemented easily. If one increases the CPU utilization bounds of EDF and RM, the system becomes stronger in hard real-time power. If one decreases them, the system becomes stronger in soft real-time power. If they are assigned properly, the system not only has some hard real-time power but also can provide soft real-time services with higher QoS.

3.2 Schedulability Analysis

Definition 1. CPU utilization factor u_i of the real-time task $t_i(c_i, P_i)$ is the ratio of c_i to P_i. c_i is the worst case computing time of task t_i and P_i is the period of task t_i if it is periodic or the relative deadline of task t_i if it is sporadic.

Assumption 1. All the tasks are independent and preemptive.

For EDF, we assume that the upper bound of its CPU utilization is U^{EDF}, and the current CPU utilization of the tasks scheduled by EDF is U'_{EDF}. When a new task adopting EDF arrives, it is acceptable, if and only if the inequality (1) is true. This can be proven by Theorem 4 in [13]. Therein, $u^{'}$ is the CPU utilization of the new task.

$$u^{'} + U'_{EDF} \le U^{EDF} \tag{1}$$

For RM, we assume that the upper bound of the CPU utilization assigned to it is U^{RM}, and the current CPU utilization of the tasks scheduled by RM is U'_{RM}. When a new task adopting RM arrives, it is schedulable, if and only if the inequality (2) is true. This can be proven by Theorem 3 in [13]. Therein, $u^{'}$ is the CPU utilization of the new task. n is the number of the tasks scheduled by RM in the system currently.

$$u^{'} + U'_{RM} \le (n+1)(2^{1/(n+1)}-1)\ U^{RM} \tag{2}$$

For SD, most of the SD scheduling algorithms are based on *General Processor Sharing* (GPS) [3, 8, 23, 25, 26]. Suppose a GPS server executes at a fixed rate U^{SD} (which is less than or equal to one), and each task t_i has a reservation ratio u_i which is a positive real number. Each task t_i is guaranteed to be served at a rate of

$$g_i = \frac{u_i}{\sum_j u_j} U^{SD} \tag{3}$$

independent of the actual workloads of other tasks. In other words, the guaranteed CPU service rate g_i for task□t_i will not be affected by the actual behavior of any t_j, $i \neq j$. On the other hand, with the guaranteed CPU service rate, a real-time task can meet all its deadlines as long as its actual workload does not exceed its reserved rate, i.e. $u_i \le g_i$.

4 Performance Evaluation

Table 1 shows the comparison of open degree about Deng's scheme [9], Wang's scheme [15] and TDPS. From Table 1 we can see our TDPS is more open than the other two schemes.

In a real-time system, an important metric of its scheduler performance is the scheduling latency. The scheduling latency is defined as the time from the occurrence of a scheduling chance to the time before the context switch for this scheduling. The lower the scheduling latency, the better the scheduler. Another important metric of its performance is the deadline missing rate, which is defined as the ratio of the number of the real-time task instances having missed their deadlines to the number of all real-time task instances. Each sporadic task is a task instance. Each period of a periodic task is a task instance.

To evaluate the performance of TDPS, we perform many tests in our real-time system with single processor P4 2.0GHz and 256MB memory.

Table 1. Comparison of open degree of three schemes

Scheme	Task modes	Parallel and distributed applications	Separating scheduling mechanism from policy	QoS control	Extensibility	Dynamic environments
Deng's	HRT/NRT	Not support	Support	no	yes	suitable
Wang's	HRT/SRT/NRT	support	Not support	no	yes	suitable
TDPS	HRT/SRT/NRT	support	Support	yes	yes	suitable

4.1 Scheduling Latency Testing

In this experiment, we sample 100 scheduling points in succession and record the scheduling latency at every scheduling point. The testing result is shown in Fig.2. Fig.3 shows the statistics of the scheduling latency of Deng's scheme vs. TDPS. From Fig.3, we can see that in the system with Deng's scheme, the minimal, maximal and average of the scheduling latency are 6, 19, and 13.17 microseconds, respectively. However, after Deng's scheme is replaced with our TDPS, the minimal, maximal and average scheduling latency decreased by 4, 5, and 3.99 microseconds, respectively. In Fig.3, 39% of the scheduling latency of TDPS is around 10 microseconds, but 35% of that of Deng's scheme is around 15 microseconds or less. This indicates that the scheduler with TDPS has lower overhead and higher efficiency than Deng's scheme under the same environment.

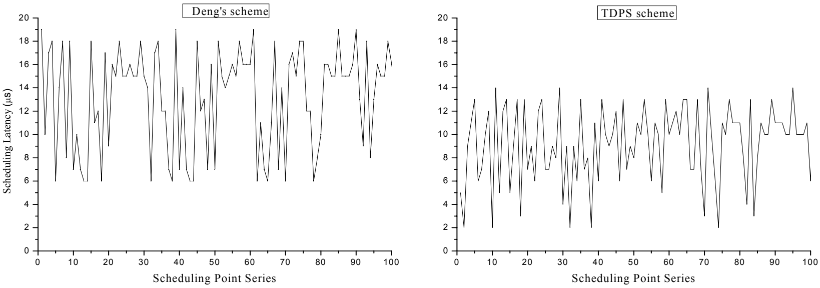


Fig. 2. The scheduling latency of Deng's scheme vs. TDPS

4.2 Deadline Missing Rate Testing

In this experiment, we remove the process of the schedulability analysis from the system so that all the tasks in the submitted applications can be accepted. All the tasks

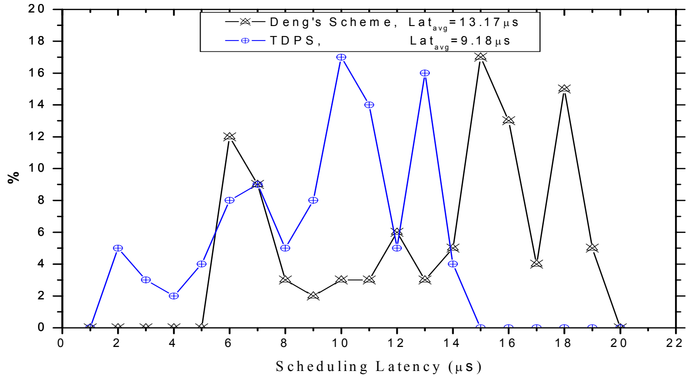


Fig. 3. The statistics of the scheduling latency of Deng's scheme vs. TDPS

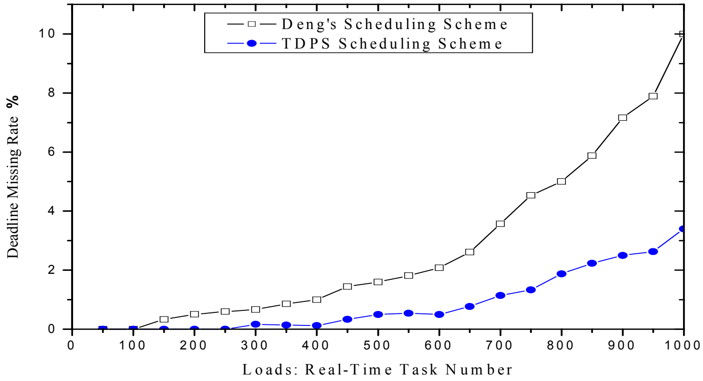


Fig. 4. The deadline missing rate of Deng's scheme vs. TDPS

in this test are hard periodic or sporadic. 1000 applications (each consists of one real-time task) are used in this experiment. We test the deadline missing rate in our real-time system with TDPS and Deng's scheme respectively under the same environment. We compute the deadline missing rates and record them under the different loads. The testing result is shown in Fig.4.

From Fig.4, we can see that the deadline missing rate of the real-time system with TDPS is lower than that of the system with Deng's scheme [9]. With the increasing of the system load, the deadline missing rate of the real-time system with Deng's scheme [9] rises rapidly than that of the system with TPDS. This indicates that TDPS has better performance and can meet the deadlines of more real-time tasks than Deng's scheme [9] under the same environment.

5 Conclusions

In this paper, we propose the *Two-Dimensional Priority Scheduling* (TDPS) for open real-time systems. In TDPS, the execution order of a task is determined not only by the task priority but also by its scheduling policy priority. In TDPS systems, the tasks with the highest scheduling policy priority are scheduled first and the tasks with the same scheduling policy are executed in the order determined by their scheduling policy. TDPS separates scheduling mechanism from scheduling policy and permits the users to choose a scheduling policy for each of the tasks in their applications.

We also introduce the CPU utilization control to enhance TDPS. The method can be used to implement different real time systems with different goals (such as hard, soft or hybrid real-time systems), which not only simplifies the schedulability analysis but also can provide the real-time services with different QoS. TDPS has high extensibility and new scheduling policies can be added to the system easily. TDPS is not only suitable for open real-time systems with single processor but also suitable for parallel and distributed real-time systems. The experimental results show that TDPS is more efficient than Deng's scheme.

References

1. A. S. Tanenbaum and A. S. Woodhull, *Operating Systems Design and Implementation*, 2nd Edition, Prentice Hall, pp.93, 1997.
2. C. L. Liu and J. Layland, "Scheduling Algorithms for Multiprogramming in a Hard Real-Time Environment", *Journal of the ACM*, Vol.20, No.1, pp.46-61, 1973.
3. A. K. Parekh and R. G. Gallager, "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case" , *IEEE/ACM Trans. Networking*, Vol.1, No.3, pp.344-357, June 1993.
4. A. Demers, S. Keshav and S. Shenker, "Analysis and Simulation of a Fair Queuing Algorithm," In: *Journal of Internetworking Research and Experience*, pp.3-26, Oct. 1990.
5. J. C. R. Bennett and H. Zhang, "WF2Q: Worst-case fair weighted fair queuing", *Proc. of IEEE INFOCOMM'96*, San Francisco, CA, pp.120-128, March 1996.
6. I. Stoica, H. Zhang, and T. S. E. Ng, "A Hierarchical Fair Service Curve Algorithm for Link-Sharing, Real-Time and Priority Services", *Proc. of ACM SIGCOMM'97*, Cannes, France, 1997.
7. M. Spuri and G. C. Buttazzo, "Efficient aperiodic service under the earliest deadline scheduling", *Proc. of the 15th IEEE Real-Time Systems Symposium*, Dec. 1994.
8. L. Abeni and G. C. Buttazzo, "Integrating multimedia applications in hard real-time systems", *Proc. of the 19th IEEE Real-Time Systems Symposium*, Dec. 1998.
9. Z. Deng, J. W-S. Liu, and J. Sun, "A Scheme for Scheduling Hard Real-Time Applications in Open System Environment", *Proc. of 9th Euromicro Workshop on Real- Time Systems*, pp.191-199, June 1997.
10. H. Kopetz, "The Time-Triggered Model of Computation", *Proc. of the 19th IEEE Real-Time Systems Symposium*, Madrid, Spain, pp.168-177, Dec. 1998.
11. C.-C. Han, K.-J. Lin, and C.-J. Hou, "Distance-constrained scheduling and its applications to real-time systems", *IEEE Trans. Computers*, Vol.45, No.7, pp.814-826, Dec. 1996.
12. Z. Deng and J. W-S. Liu, "Scheduling Real-Time Applications in Open System Environment", *Proc. of 18th IEEE Real-Time Systems Symposium*, San Francisco, CA, Dec. 1997.
13. T.-W. Kuo and C.-H. Li, "A fixed-priority-driven open environment for real-time applications", *Proc. of the 20th IEEE Real-Time Systems Symposium*, pp.256-267, Dec. 1999.
14. T.-W. Kuo, K.-J. Lin, and Y.-C. Wang, "An open real-time environment for parallel and distributed systems", *Proc. of 20th International Conference on Distributed Computing Systems*, pp.206-213, April 2000.

15. Y. C. Wang and K. J. Lin, "Implementing a general real-time scheduling framework in the RED-Linux real-time kernel", *Proc. of the 20th IEEE Real-Time Systems Symposium*, pp.246-255, Dec. 1999.
16. K. J. Lin and Y. C. Wang, "The design and implementation of real-time schedulers in RED-linux", *Proc. of the IEEE*, Vol.91, No.7, pp.1114-1130, July 2003.
17. S. A. Brandt, S. Banachowski, C. Lin, and T. Bisson, "Dynamic integrated scheduling of hard real-time, soft real-time and non-real-time processes", *Proc. of the 24th IEEE Real-Time Systems Symposium*, pp.396–407, Dec. 2003.
18. S. Goddard and L. Xu, "A variable rate execution model", *Proc. of the 16th Euromicro Conference on Real-Time Systems*, pp.135–143, July 2004.
19. G. C. Buttazzo, G. Lipari, M. Caccamo, and L. Abeni, "Elastic scheduling for flexible workload management", *IEEE Transactions on Computers*, 51(3):289–302, Mar. 2002.
20. S. Banachowski, T. Bisson, and S. A. Brandt, "Integrating best-effort scheduling into a real-time system", *Proc. of the 25th IEEE Real-Time Systems Symposium*, pp.139-150, Dec. 2004.
21. L. Marzario, G. Lipari, P. Balbastre, and A. Crespo, "IRIS: A new reclaiming algorithm for server-based real-time systems", *Proc. of 10th IEEE Real-time and Embedded Technology and Applications Symposium*, pp.211-218, May 2004.
22. G. Lipari and G. C. Buttazzo, "Scheduling real-time multitask applications in an open system", *Proc. of the 11th Euromicro Conference on Real-Time Systems*, June 1999.
23. M. Spuri and G. Buttazzo, "Scheduling Aperiodic Tasks in Dynamic Priority Systems", *Proc. of the 17th IEEE Real-Time Systems Symposium*, pp.179-210, Dec. 1996.
24. http://www.rtai.org/.
25. I. Stoica, H. Abdel-Wahab, K. Jeffay, S. K. Baruah, J. E. Gehrke, and C. G. Plaxton, "A Proportional Share Resource Allocation Algorithm for Real-Time, Time-Shared Systems", *Proc. of the 17th IEEE Real-Time Systems Symposium*, pp.288-299, 1996.
26. T.-W. Kuo, W.-R. Yang, and K.-J. Lin, "EGPS: A Class of Real-Time Scheduling Algorithms Based on Processor Sharing", *Proc. of the 10th Euromicro Workshop on Real-Time Systems*, pp.27-34, June 1998.

An Enhanced Dynamic Voltage Scaling Scheme for Energy-Efficient Embedded Real-Time Control Systems

Feng Xia and Youxian Sun

National Laboratory of Industrial Control Technology,
Zhejiang University, Hangzhou 310027, China
{xia, yxsun}@iipc.zju.edu.cn

Abstract. Real-Time Dynamic Voltage Scaling (RT-DVS) has been one of the most important techniques for energy savings in battery-powered embedded systems. However, pure RT-DVS approaches rarely take into account the actual performance requirements of the target applications. With the primary goal of further reducing energy consumption while satisfying Quality of Control (QoC) requirements in real-time control systems, an enhanced dynamic voltage scaling (EDVS) scheme is suggested. Following the direct feedback scheduling methodology, EDVS exploits a QoC-aware adaptive resource allocation mechanism. It enables flexible timing constraints on control tasks, which facilitates further energy saving over pure RT-DVS. Simulation experiments argue that EDVS is highly cost-effective and can save much more energy over the optimal pure RT-DVS scheme, while providing comparable QoC.

1 Introduction

In recent years, dynamic voltage scaling (DVS) [1,2], which exploits multiple voltage and frequency levels to reduce CPU energy consumption of embedded and real-time systems, has attracted a lot of attention from both academic and industrial communities. In particular, a large body of DVS work exists for real-time applications running on single processors [3]. Most RT-DVS (Real-Time DVS) algorithms, such as Cycle-Conserving and Look-Ahead from [1] as well as DR-OTE and AGR from [2], generally adjust the supply voltage and clock frequency with respect to workload variations so as to reduce energy consumption under the task schedulability constraint. They have proved to be capable of providing significant energy savings for embedded processors while still meeting the task deadlines.

As an important subclass of embedded systems, battery-powered embedded real-time controllers must operate in energy-efficient fashion while guaranteeing required Quality of Control (QoC) [4]. However, little work is dedicated to real-time control tasks in the literature of DVS. Most state-of-the-art RT-DVS algorithms rarely take into account the resulting performance of target applications other than real-time guarantees when determining the voltage level of the processor. These algorithms are usually based on fixed timing constraints such as periods and deadlines of real-time tasks. That is, they typically derive the processor speed providing timeliness guarantees during run time according to pre-specified periods/deadlines of the task set, and

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 539 – 548, 2006.

the timing attributes will never be intentionally changed, e.g. in response to the actual QoC requirements, for the sake of energy savings. In a multitasking control system, however, the controlled plants may experience various perturbations in spite of successful schedule of the whole control task set. Although shorter sampling periods may be preferable when reacting to perturbations in order to improve the control performance, they imply waste of resources (i.e., CPU time and energy) when the system is in steady state [5-7]. This feature of real-time control applications makes it possible to dynamically allocate resources to each control task according to their real demands. Since fixed timing constraints are commonly used when dealing with control tasks, not surprisingly, existing RT-DVS systems perform bad both in control performance improvement when the system is in transient process and in energy saving when the system is in steady state.

In this paper, we consider a set of controller tasks that run on the same energy-limited embedded CPU. Our goal is to enhance the performance of RT-DVS by further reducing energy consumption without jeopardizing control performance. As a substantial step towards the integration of real-time control and power-aware computing, we present an enhanced dynamic voltage scaling (EDVS) scheme. Based on system-level pure DVS algorithms, a QoC-aware resource allocation mechanism following the direct feedback scheduling methodology [8-10] is explored to dynamically adjust the (sampling) period of each control task with respect to the instantaneous performance of the target plant. As a general rule, a control loop will get a small period when experiencing disturbance and a large period in steady state. In contrast to the traditionally employed fixed timing constraints that inevitably limit the efficiency of DVS algorithms, the proposed QoC-aware resource allocation mechanism enables flexible timing constraints on control tasks, which allows allocating resources to control loops according to their current performance requirements. By exploiting this application adaptation, further energy savings over pure DVS algorithms are expected to achieve.

The use of flexible timing constraints on real-time control tasks is suggested in [5]. In [6], Martí *et al.* present a state feedback based optimal resource allocation policy that maximizes control performance within constrained resources for multi-loop control systems. Velasco *et al.* [7] present a dynamic approach to bandwidth management in networked control systems that allow control loops to consume bandwidth according to the dynamics of the controlled process. However, none of them consider the energy consumption of the processor. As far as we know, there is only one work by H. S. Lee and B. K. Kim [11] that deals with energy management in real-time control systems. However, the dynamic solution in the paper does not assign sampling periods of control tasks using a particular direct feedback scheduling algorithm as we do. In contrast to only two period values for each control task in [11], the sampling periods we use can vary continuously in an allowable range.

The remainder of this paper is structured as follows. In Section 2, we describe an optimal pure RT-DVS mechanism as a baseline for comparison with our EDVS. Section 3 illustrates the framework of EDVS and develops the relevant algorithm for control periods adjustment. The performance of EDVS is evaluated in Section 4. Section 5 concludes this paper.

2 The Optimal Pure RT-DVS

We here consider a battery-powered embedded microcontroller that is responsible for running n independent controller tasks $\{T_\mathrm{i}\}$ concurrently. Each task T_i has a period h_i equal to its relative deadline. The execution time of T_i under the maximum operating voltage/frequency level is given by C_i. We assume that the voltage/frequency of the CPU could be adjusted continuously with a scaling factor α ($0 < \alpha \leq 1$), where α=1 implies that the processor operates at the full speed (maximum voltage level). In the following, we will use α to denote the normalized voltage level or processor speed. It is worthy noting that although we have assumed continuous voltage/frequency levels, the proposed EDVS scheme, with only minor extensions, is also applicable to real processors with discrete voltage levels. When the voltage/frequency is rescaled with α, the actual execution time of T_i will be C_i/α. In addition, the switching overheads between voltage/frequency levels are neglected.

Throughout this paper, we adopt the simple model in [12] to estimate the normalized energy consumption of CPU as $E(\alpha) = \alpha^2$. Since the energy model is a monotonic increasing function of the voltage scaling factor, α is expected to be minimized in favor of maximum energy saving.

With a pure RT-DVS scheme, the processor speed is typically calculated such that the CPU utilization is fully exploited by slowing down task execution while maintaining task schedulability [1]. We assume that the task set is scheduled based on EDF (Earliest-Deadline-First) algorithm. Thus the schedulability condition is $\sum C_i / \alpha h_i \leq 1$ [13]. In this paper, we employ a representative system-level RT-DVS algorithm, denoted pDVS (pure DVS), for optimal voltage scaling for the system described above, assuming fixed timing constraints. This algorithm will also be used as a baseline for the illustration and evaluation of our EDVS scheme.

The pDVS approach works in an interval-based manner, i.e., it performs the voltage scaling task at regular intervals at run time. It obtains the minimum possible processor speed, which leads to the maximum energy saving, as follows.

$$\alpha_{\min} = \sum C_i / h_i \tag{1}$$

Although pDVS is simply described, it has been found [12] that this scheme is optimal for the system considered. That is, for the above system under fixed timing constraints, pDVS achieves the maximum energy saving while guaranteeing the system schedulability with EDF.

3 Enhanced Dynamic Voltage Scaling

In this section, we present EDVS, a novel methodology to enhance the performance of pure RT-DVS schemes using a QoC-aware resource allocation mechanism that exploits direct feedback scheduling. The framework of EDVS is illustrated. It is constructed upon pDVS and flexible timing constraints on control tasks. The sampling period of each control task is dynamically adjusted with respect to instantaneous control performance in order to further improve the energy efficiency over pDVS. The algorithm used for QoC-aware resource allocation is also given.

3.1 Framework

According to digital control theory [14], the sampling period of the controller generally could be varied in certain ranges while still providing satisfactory control performance. From the system and energy models used, it could be found that the CPU energy consumption decreases when the (sampling) periods of control tasks increase, assuming constant task execution times and steady CPU utilization. Intuitively, shorter sampling periods yield better control performance. While it is desirable to execute a task with the highest rate when the target plant experiences perturbations, the sampling period may be lengthened without losing the specified performance when the system is in steady state [5]. There are successful examples that utilize this feature of control systems to dynamically manage the system resources such as CPU time [6] and network bandwidth [7]. We here build our EDVS based on this observation of flexible timing constraints and aim to achieve further energy savings over pDVS.

The framework of EDVS is given in Fig. 1. Compared to pure RT-DVS schemes, an additional *direct feedback scheduler* is introduced in EDVS. The functionality of the pDVS is just the same as what has been described in Section 2. The role of the direct feedback scheduler is to dynamically adjust the sampling period of each task respectively. We use the notion of *direct* feedback scheduler here to indicate that it determines the sampling period of each control task *directly* according to the instantaneous control performance of the relevant loop. In this way, EDVS provides a QoC-aware dynamic resource allocation mechanism, which in turn realizes application-adaptive energy management. The direct feedback scheduler works in a periodic manner with the same invocation interval of the pDVS component. The algorithm used in the direct feedback scheduler will be detailed below.

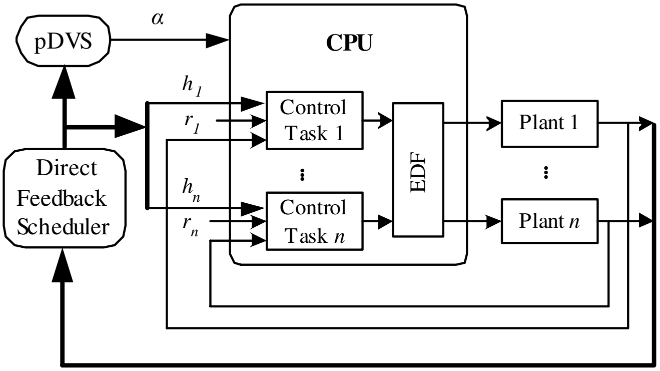


Fig. 1. Framework of EDVS

3.2 QoC-Aware Resource Allocation

A prerequisite for online assignment of sampling periods using direct feedback scheduling is selecting a proper metric to indicate instantaneous control performance. A natural and quite reasonable choice is the absolute system error (denoted *err*), which

which is defined as the absolute difference between the plant output and the reference of the control loop. In general, the larger the error, the more critical the loop. However, when the plant output sharply oscillates around the reference, the system error might still be very small sometimes, which could not reflect the real control performance and requirements. Therefore, we define the following instantaneous performance index

$$ind_i(k) = \lambda \cdot ind_i(k-1) + (1-\lambda) \cdot err_i(k) \qquad i = 1, ..., n \tag{2}$$

where λ is a forgetting factor, and k is the invocation instant of EDVS. Next, we examine how to determine a proper sampling period for each control task τ_i based on the current value of ind_i. Since the feedback scheduler assigns each sampling period respectively, we will omit the subscript i from all variables wherever possible in the following for the sake of simple description.

To guarantee closed-loop stability, any control system has an upper bound on the sampling period. Using digital control theory [14], this upper bound h_{max} could be easily obtained. Additionally, there is also a lower bound h_{min} on the allowable periods for each task, which comes from the task schedulability constraint. A theoretical value of h_{min} can be obtained from $C_i / h_{\min,i} + \sum_{k \neq i} C_k / h_{\max,k} = 1$. However, this h_{min} value is too optimistic and prone to incur overloads even if the CPU operates at the highest voltage level all the time. Therefore, we here select a larger h_{min} for each task such that $\sum C / h_{\min} \leq U_{ref}$. For simplicity, we assume the initial sampling period $h_0 = h_{min}$. The algorithm used to determine the new sampling period of each control task is given by

$$h(k) = \eta(k) \cdot h_0 = \eta(k) \cdot h_{\min} \tag{3}$$

where η is the *period rescaling factor* defined as

$$\eta(k) = \begin{cases} h_{\max} / h_{\min} & (ind(k) \leq err_{\min}) \\ \dfrac{h_{\max} / h_{\min} - 1}{e^{-\beta \cdot err_{\min}} - e^{-\beta \cdot err_{\max}}} (e^{-\beta \cdot ind(k)} - e^{-\beta \cdot err_{\max}}) + 1 & (err_{\min} < ind(k) < err_{\max}) \\ 1 & (ind(k) \geq err_{\max}) \end{cases} \tag{4}$$

where β is a constant introduced to enhance the effect of exponential function.

According to (4), the sampling period will be directly set at the minimum once *ind*(*k*) exceeds a certain limit err_{max}, which indicates high criticality. The objective of this operation is to improve the system state as soon as possible. In contrast, the period will be the maximum if *ind*(*k*) becomes less than another limit err_{min}, which implies that the system approaches a steady state. We set the maximum period to achieve the largest possible energy saving. In other cases, a period that decreases exponentially as the filtered control error increases will be assigned. The reason for using an exponential function is to reflect the highly increasing resource requirements of the loop where the system output significantly deviates from the reference. Several example curves of the period rescaling factor with different β values are depicted in Fig. 2, where the x-axis is *ind* and the y-axis is η.

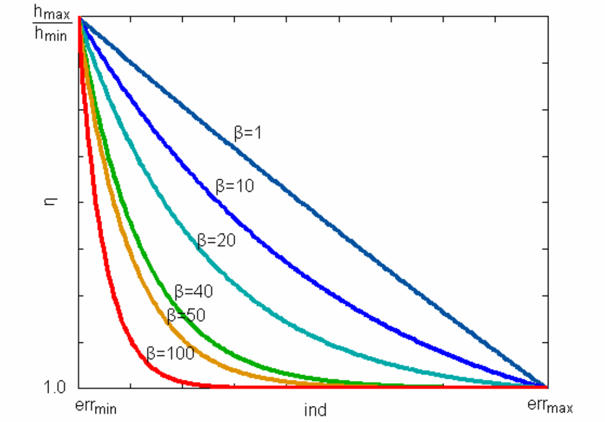


Fig. 2. Illustration of function η

Remark 1. Adjusting the sampling period at run-time will introduce sampling period jitters in a control loop, which could significantly degrade control performance. Fortunately, one could compensate for sampling period jitters in the controller, and many such methods have been developed in the control community. Therefore, in order to eliminate the impact of sampling period jitters that derives from the direct feedback scheduling on the performance of EDVS, it is necessary that the control algorithm for each loop is designed capable of online compensating for these jitters.

To summarize, the EDVS operates as follows. Upon every invocation of the direct feedback scheduling algorithms, the output of each controlled plant is sampled and the absolute control error err_i is calculated. Based on this value, a new sampling period will be determined by the direct feedback scheduler for each control task respectively. These new sampling periods will then be used in the pDVS algorithm to dynamically adjust the operating speed of the processor. Accordingly, the pseudo code for the EDVS scheme is as follows.

```
//At every invocation instant
Procedure EDVS {
   FOR each control loop i
      Sample the plant output $y_i$;
      $err_i$← abs($y_i$-$r_i$);
      //$r_i$: the reference of loop i
      Compute $ind_i$ using Eq. (2);
      Compute a new sampling period using Eq. (3) and (4);
      //Update the sampling period
   END
   pDVS {
      Compute $\alpha$ using Eq. (1), and perform voltage scaling.
   }

}
```

As we can see, EDVS is indeed an improved scheme over pure DVS, and it can operate upon any appropriate DVS mechanism other than pDVS, although we employ pDVS in the description.

4 Performance Evaluation

In this section, four independent plants with the same model $G(s)=1000/(s^2+s)$ are considered. The corresponding controllers, which execute PID (Proportional-Integral-Derivative) algorithms, are well-designed respectively. In order to compensate for sampling period variations, the controller parameters can be online updated. The timing parameters (C, h_{min}, h_{max}) of four control tasks are given as (2, 10, 17), (2, 9, 17), (2, 8, 17), and (2, 7, 16) in time unit of *ms*. Recall that $h_0 = h_{min}$ for all tasks. During all simulations, the nominal task execution times remain constant. The invocation interval for DVS is chosen as 40 ms. Some parameters in EDVS are set as follows: $\lambda = 0.3$, $err_{max} = 0.2$, and $err_{min} = 0.02$. Using this setup, we have conducted extensive simulation experiments based on Matlab/TrueTime [8] to evaluate the performance of EDVS in different scenarios.

To record the control performance of each loop, we employ the generally used performance index IAE (Integral Absolute Error) [14]. It is calculated as a cost function $J_i = \sum_k | y_i(k) - r_i | \cdot h_i$, where y_i and r_i are the output and reference of each plant respectively. Note that the higher the cost, the worse the QoC.

4.1 Scenario 1: Dynamic Task Activation

In this set of simulations, we assess the performance of EDVS under significant workload variations induced by dynamic task activations. According to [12], pDVS is optimal in energy saving among pure RT-DVS algorithms for the system we consider. Therefore, we compare our EDVS with this optimal pure DVS.

Simulation experiments run as follows. At time t = 0, only T_1 is on. T_2 becomes active at time t = 1s. At t = 2s, T_3 and T_4 are switched on. Each control task experiences an input step change (i.e. the only type of perturbations in control loops) at the start of execution respectively. Another input step change is issued on each control task simultaneously at t = 3s. The whole experiment lasts 4 seconds. This pattern is repeated for three different approaches: I) NON-DVS: no voltage scaling, CPU always operates at the maximum voltage level, II) pDVS: the optimal pure RT-DVS algorithm, and III) EDVS: our scheme with $\beta = 40$.

The normalized energy consumptions for different approaches considered are depicted in Fig. 3. Without any voltage scheduling, the normalized energy consumption of NON-DVS remains one all the time. As can be seen, significant energy savings are achieved using the pDVS scheme compared to the NON-DVS case, especially when the workload is light, e.g. from time t = 0 to 2s. Throughout the experiment, EDVS always consumes the least energy. It saves up to 68.9% more energy in comparison with pDVS. The average energy consumption of pDVS is 50.9%, while that of EDVS is only 26.3%. It is clear that there is a 24.6% average additional energy saving.

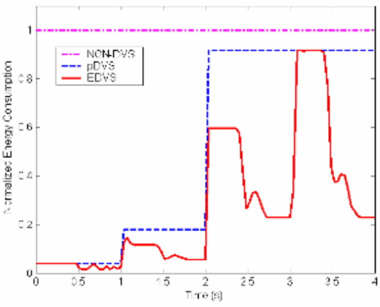


Fig. 3. Normalized energy consumptions under different schemes

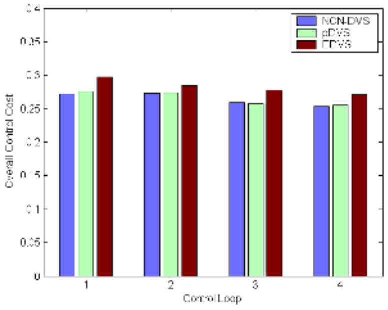


Fig. 4. Control costs of different loops under different schemes

The overall control costs for different approaches are given in Fig. 4. Always with the highest execution rates (the highest voltage level), each loop performs the best in the NON-DVS case. Although the execution of control tasks may be slowed down, the control performance under pDVS is nearly the same as that of NON-DVS. Comparing EDVS with pDVS, one could find that for all four loops, the performance degradation under EDVS is considerably slight. The overall control costs of four loops only increase 7.6%, 4.0%, 8.2%, and 6.3%, respectively. From the system responses of the four loops, we also find that the difference of control performance between EDVS and pDVS is quite minor.

In summary, EDVS is highly cost-effective to improve the energy-efficiency of embedded real-time control systems where the workload may vary significantly. It can easily achieve more additional energy savings over the optimal pure RT-DVS scheme at the expense of only minor QoC degradation.

4.2 Scenario 2: Different β Values

Since different β values result in different η function curves, as shown in Fig. 2, it is obvious that the choice of β influences the performance of EDVS. Therefore, we simulate EDVS with different β values in this set of experiments. The simulation pattern remains the same as in scenario 1. The set of β values is chosen as {1, 10, 20, 40, 50, 100, ∞}, where $\beta \rightarrow \infty$ implies a sampling period adjustment mechanism similar to the dynamic solution in [11].

Fig. 5 depicts the CPU energy consumptions under EDVS with different β values. The average normalized energy consumption (denoted E_{AVG}) in each simulation is summarized in Table 1. It can be seen that energy saving increases with decreased β values. In addition, in the extreme case with an infinite β, EDVS can still achieve an average additional energy saving of 18.7% over the optimal pure DVS, i.e., pDVS.

As shown in Fig. 6, the difference between the control costs of four loops under different β choices is not so significant. With the β values we have chosen, the

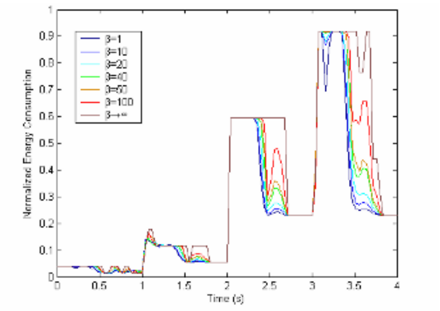


Fig. 5. Normalized energy consumptions under EDVS with different β values

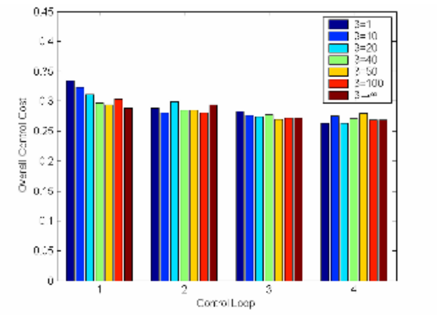


Fig. 6. Control costs under EDVS with different β values

Table 1. Average energy consumption and average control cost under different β values

	β=1	β=10	β=20	β=40	β=50	β=100	$\beta\to\infty$
E_{AVG}	23.5%	24.3%	25.1%	26.3%	26.8%	28.6%	32.2%
J_{AVG}	0.292	0.289	0.287	0.283	0.282	0.282	0.281

average control costs of four loops, $J_{AVG} = \sum J_i / n$, are given in Table 1. As the β value increases, the average control cost decreases, which implies better QoC. Obviously, the case $\beta\to\infty$ yields the best overall QoC. Although β=1 yields the worst QoC, the control responses are still quite satisfactory.

It can be outlined from Table 1 that decreasing β leads to higher energy efficiency yet worse QoC. In this sense, there is a trade-off between large energy saving and high QoC when determining the proper value for β at design time.

5 Conclusions

This paper suggests a novel QoC-aware scheme to enhance the energy-saving capability of dynamic voltage scaling in the context of embedded real-time control. Based on pure RT-DVS, the proposed EDVS scheme exploits a direct feedback scheduling mechanism. It enables flexible timing constraints on control tasks and control aware resource allocation. Through dynamically adjusting the (sampling) periods of control tasks according to instantaneous control performance requirements, EDVS allows further improving energy efficiency over existing pure RT-DVS approaches. This is a substantial step towards the integration of real-time control and power-aware computing techniques. Simulation results argue that EDVS can easily save much more energy over the optimal pure RT-DVS scheme, while guaranteeing good control performance.

Acknowledgement. This work is partially supported by Key Technologies R&D Program of Zhejiang Province (No. 2005C21087), Academician Foundation of

Zhejiang Province (No. 2005A1001-13), and Specialized Research Fund for the Doctoral Program of Higher Education (No.20050335020).

References

1. P. Pillai and K.G. Shin: Real-Time Dynamic Voltage Scaling for Low Power Embedded Operating Systems. In: Proc. 18th Symp. Operating System Principles (2001) 89-102
2. H. Aydin, R. Melhem, D. Mosse, and P. Mejia-Alvarez: Power-Aware Scheduling for Periodic Real-Time Tasks. IEEE Trans. Computers 53:5 (2004) 584-600
3. N.K. Jha: Low-power system scheduling, synthesis and displays. IEE Proc.-Comput. Digit. Tech. 152:3 (2005) 344-352
4. Feng Xia, Xiaohua Dai, Xiaodong Wang, and Youxian Sun: Feedback Scheduling of Real-Time Control Tasks in Power-Aware Embedded Systems. In: Proc. 2nd Int. Conf. on Embedded Software and Systems, Xi'an, China, IEEE CS Press (2005) 513-518
5. P. Martí, G. Fohler, K. Ramamritham, J. M. Fuertes: Improving quality-of-control using flexible time constraints: Metric and scheduling issues. In: 23nd IEEE RTSS, Austin, TX (2002) 91-100
6. P. Marti, C. Lin, Scott Brandt, M. Velasco, J.M. Fuertes: Optimal State Feedback Based Resource Allocation for Resource-Constrained Control Tasks. In: 25th IEEE RTSS, Lisbon, Portugal (2004) 161-172
7. M. Velasco, J. Fuertes, C. Lin, P. Marti, S. Brandt: A Control Approach to Bandwidth Management in Networked Control Systems. In: 30th IEEE IECON, Busan, Korea (2004) 2343-2348
8. K.-E. Årzén, A. Cervin, D. Henriksson: Resource-Constrained Embedded Control Systems: Possibilities and Research Issues. In: Co-design of Embedded Real-Time Systems Workshop, Porto, Portugal (2003)
9. Feng Xia, Xiaohua Dai, Zhi Wang, and Youxian Sun: Feedback Based Network Scheduling of Networked Control Systems. In: Proc. 5th IEEE ICCA, Budapest, Hungary (2005) 1231-1236
10. Feng Xia, Xiaohua Dai, Youxian Sun, and Jianxia Shou: Control Oriented Direct Feedback Scheduling. International Journal of Information Technology (to appear)
11. H. S. Lee, B. K. Kim: Dynamic Voltage Scaling for Digital Control System Implementation. Real-Time Systems 29 (2005) 263-280
12. A. Sinha and A. P. Chandrakasan: Energy efficient real-time scheduling. In: Proc. IEEE/ACM Int. Conf. Computer Aided Design (2001) 458-463
13. C. Liu and J. Layland: Scheduling Algorithms for Multiprogramming in a Hard Real-Time Environment. J. ACM 20 (1973) 46-61
14. K.J. Åström and B. Wittenmark: Computer-Controlled Systems. Prentice Hall (1997)

Adaptive Load Balancing Mechanism for Server Cluster[*]

Geunyoung Park[1], Boncheol Gu[1], Junyoung Heo[1], Sangho Yi[1], Jungkyu Han[1], Jaemin Park[1], Hong Min[1], Xuefeng Piao[1], Yookun Cho[1], Chang Won Park[2], Ha Joong Chung[2], Bongkyu Lee[3], and Sangjun Lee[4]

[1] Seoul National University
{gypark, bcgu, jyheo, shyi, jkhan, jmpark, hmin, hbpark, cho}@ssrnet.snu.ac.kr

[2] Intelligent IT System Research Center, Korea Electronics Technology Institute
{parkcw, chunghj}@keti.re.kr

[3] Cheju National University
bklee@venus1.cheju.ac.kr

[4] Soongsil University
sjlee@computing.ssu.ac.kr

Abstract. Server cluster provides high availability, scalability, and reliability by gathering server nodes into a group. Client requests need to be distributed to each server node fairly to maximize the performance of server cluster. In this paper, we propose an adaptive and efficient load balancing algorithm for the server cluster. The proposed algorithm computes the load of server nodes with the usages of computer resources and their weights. These weights are determined dynamically based on the statistics of the usages. The experimental result shows that the proposed algorithm can prevent the bottleneck of server cluster efficiently compared with existing algorithms. This guarantees its adaptability even though there are changes to the characteristic of service.

1 Introduction

With the improvement of network and hardware, Internet bandwidth is growing continuously. Furthermore, as there is greater demand for multimedia data such as images, audio, and video through HTTP service, there is a need for storage with large capacity for storing multimedia data.

Storage capacity can be easily increased by replacing or upgrading the hardware, but it is costly, may overload the server, increase response time, makes it difficult to properly manage or facilitate data sharing, etc. Therefore, the Clustering System was developed to combine several servers into a group.

With the Clustering System, high availability and scalability could be achieved. However, there is still one remaining issue to be resolved: how the requests from clients to servers can be distributed properly? This is the problem of load balancing. Failure of balancing results in the uneven distribution of client requests. This

[*] This research was supported by the Brain Korea 21 project.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 549–557, 2006.

in turn results in the degradation of the processing capability of the whole system as well as the ineffective usage of resources. Therefore, we need a load balancer to address this problem.

The load balancer is located on the front-end of the server cluster and all the requests from clients arrive at this load balancer first. After receiving the requests, the load balancer distributes them to the servers of the cluster. Hence, the load balancer contributes to enhance the performance of the system.

Linux Virtual Server (LVS)[1, 2] is a remarkable clustering system with a load balancer. LVS is an open source project for the virtual server system based on Linux. LVS consists of a load balancer, a pool of servers, and back-end storage system. The whole system is transparent to users. The load balancer of LVS receives requests from clients and assigns the requests to each server, which then processes the request and send responses to users directly.

The load balancer uses scheduling algorithm to determine how the client requests are to be distributed. The performance of a LVS system is dependent on the type of scheduling algorithm used. There are static scheduling algorithms such as Round-Robin and Least-Connection provided by LVS, but these algorithms cannot measure the exact loads of the server nodes due to the lack of information exchange among them.

To setup the reliable criterion according to the real load information of server nodes, dynamic scheduling algorithms such as Agent-Based Sophisticated and Scalable Scheduling Method (ABSS)[3] and Agent-based Sophisticated Scheduling Mechanism by Using Memory Utilization (ASSUM)[4] were developed. In these algorithms, the load balancer communicates with server nodes to detect the real load information.

However, there are some issues relating to the cluster system to ensure that it performs effectively in a variable environment where the characteristic of service may change at anytime. Thus, in this paper, the algorithm adaptable to such variable environments is proposed.

2 Related Works

2.1 Distributed Web Server System

Distributed web server system is composed of several web server hosts deployed in LAN or WAN, using the mechanism to distribute the client requests to each server[5]. To construct distributed web server system, there have been approaches that assure transparency for load balancing to users and make the system appear to be a single host to the outside world. Based on the entity that distributes the incoming requests among the servers, In [5], Cardellini el al. classifies the web clustering approaches into 4 groups: Client-based approach, DNS-based approach, Dispatcher-based approach, and Server-based approach.

In the client-based approach, the requests from the clients can be distributed by web clients deployed on the client systems in the form of an application. Web clients can actively route requests to the replicated servers within a cluster. The web client selects a server node of the cluster which can respond to the client's request. Then, it submits the request to the selected server node. Even though this approach provides scalability and availability, it requires the client to have the knowledge about the server cluster. Smart clients[6] are the example of client-based approach.

In the DNS-based approach, the URL-level virtual interface is used to assure transparency to clients mapping URL to IP address. The cluster DNS which is the authoritative DNS server for the system can implement multiple policies for selecting the appropriate server and spread client requests. However, many intermediate name servers and web client browsers can cache the URL-to-IP-address mapping to reduce network traffic. In this case, it is possible that the cached server nodes may receive more requests than those not cached. Moreover, the cluster DNS is not aware of the state of sever nodes, so it may route the client requests to the crashed server nodes. The examples of DNS-based approach are shown in [7] and [8].

In the dispatcher-based approach, a network component of the web server system acts as a dispatcher to centralize request scheduling and completely control client-request routing. Request routing among servers is transparent, and unlike DNS-based architectures, which deal with addresses at the URL level, the dispatcher has a single, virtual IP address (IP-SVA).

On the other hand, server-based approach uses a two-level dispatching mechanism. The primary DNS of the web system initially assigns client requests to the web server nodes; then, each server can reassign the received request to other server. Unlike the DNS-based and dispatcher-based centralized solutions, the distributed scheduling approach allows all servers to participate in load balancing of the system through the request reassignment mechanism. Through this mechanism, the system prevents a particular component of the system, such as a dispatcher, from being a bottleneck.

In this paper, we will deal the scheduling methods based on the dispatcher-based approach, which is used to centralize request scheduling and completely control client-request routing.

2.2 Scheduling Algorithms

Most of the current approaches that are used to construct distributed web server clusters perform the scheduling through the static scheduling algorithm such as random selection or round-robin, in which there is no exchange of any information between the load balancer and each server node. These static scheduling algorithms have a limit in conducting load balancing properly. Therefore, to overcome this limitation, the dynamic scheduling algorithms that route client requests according to the information from server nodes were proposed.

Static scheduling algorithms include Round-Robin, Weighted Round-Robin, Least-Connection, and Weighted Least-Connection [1]. Round-Robin assigns the requests to servers in the round robin manner. Least-Connection routes the request to the server which has the least connection. Weighted Round-Robin and Weighted Least-Connection gives weight to each server node. A drawback of these static scheduling algorithms, however, is that they are unable to reflect the actual loads of server nodes.

The Dynamic Scheduling algorithms, in which server nodes communicate the load information to the load balancer, were developed in order to address above problem. In ABSS[3] and ASSUM[4], the "Agent"s are deployed in the load balancer node and server nodes. The load balancer gathers the load information of each server node through the agents and records them on its internal "Load Information Table". Based on this information, the load balancer assigns the requests from clients to the server that is least loaded. In ABSS, the CPU usage is considered as the load information, whereas in ASSUM, the status of the memory used to manage the current socket connections with clients is considered as the load information.

3 Proposed Scheduling Method

ABSS and ASSUM are representative algorithms that use agents to measure the load information of server nodes. However, they use only single metric such as CPU usage (ABSS) and network I/O (ASSUM). Therefore, they cannot guarantee efficient scheduling when a dynamic change occurs in the tasks between CPU-bound and I/O-bound. In this section, we propose an effective load balancing algorithm that takes into account multiple load factors.

3.1 System Model

In the dispatcher-based approach, a server cluster consists of a load balancer node and several server nodes that process user requests. The whole cluster can be regarded as one virtual server by users. The load balancer node receives a request from a user, chooses server node to process the request, and forwards the request to the node selected.

3.2 Factors Used for Computing the Load

For efficient load balancing, it is important to estimate the current load state of each node in the cluster. In this paper, similarly to previous works, we consider the following factors for computing the load.

- CPU usage - CPU utilization in a period
- Memory Usage - The amount of free memory
- Number of processes - The number of processes in the ready-queue of OS.
- Number of I/O - The number of I/O operations which have been, or shall be performed
- Local Storage - The amount of free and available storage
- Network I/O usage - The ratio of network bandwidth which are used. through network adapter

Measuring each and every factor and computing the load repeatedly can incur significant overhead. To address this problem, we will select some major factors and assign weights to them based on the characteristics of services.

3.3 Computing the Load of a Node

In our proposed algorithm, we compute the load of each server node by summing up the weighted factors. At a certain point of time, the load L_{n_i} of the server node n_i can be computed as follows:

$$L_{n_i} = \sum_{f \in FactorSet} W_f V_{f,n_i}$$

where

$$\begin{aligned} L_{n_i} &= \text{The current load of node } n_i \\ FactorSet &= \left\{ f \;\middle|\; f \;:\; \begin{array}{c} \text{factors representing the load of node} \\ \text{e.g. CPUusage, MemoryUsage, I/O} \end{array} \right\} \\ W_f &= \text{The weight of factor } f \text{ . } 0 \leq W_f \leq 1 \text{ , and } \sum_{f \in FactorSet} W_f = 1 \\ V_{f,n_i} &= \text{The value based on } \frac{\text{current usage}}{\text{capacity}} \text{ of factor } f \text{ for node } n_i \end{aligned}$$

It is important to assign a proper weight to each factor. These weights can vary depending on the characteristic of the current service. In addition, assigning higher weight to the factor which is major bottleneck can achieve more efficient load balancing.

To reveal the effects of weights on the performance of the server cluster, an experiment was performed using LVS. Figure 1 shows the effectiveness of load balancing when fixed weights are assigned. The x-axis represents the rate of user request and the y-axis the rate of server response. “W(cpu)=1.0” means that the weight of cpu usage is 1.0 and all the other weights are set to 0. “W(NetI/O)=1.0” means that the weight of the amount of packets transmitted via network is 1.0. “CPU-bound service” and “I/O-bound service” show the characteristics of services. According to Fig. 1, if the weights of factors are fixed, the performance of the whole cluster can vary significantly according to the changes in the characteristic of service.

Server node overload can occur depending on the service requested by clients. For example, CPU is the main factor of a bottleneck when CGI service is provided, and network I/O becomes a bottleneck when the cluster provide a file storage service. If all the weights are fixed, we can observe the following disadvantages:

Inconvenience. The administrator of the server must find the most appropriate value for each weight, taking into account the characteristic of the service.

Inadaptability. If the server provides more than one services at the same time, it cannot determine which service is to be provided a point of time. Therefore, even if the weights are optimized for a specific service, they are not optimal, or in fact be a worse choice for another service. And hence may not be able to demonstrate its best performance.

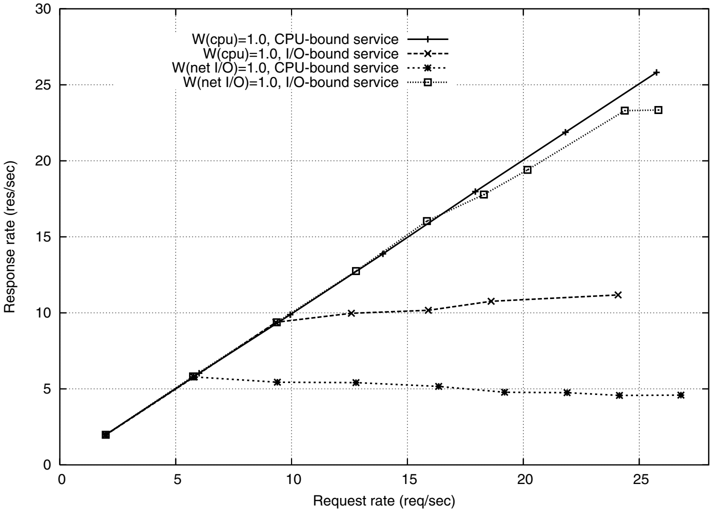


Fig. 1. Static assignment of weights

3.4 Adaptive Weight Assignment

To overcome such disadvantages, we propose a method with which the weights of factors can be adjusted in run-time. Each weight of a factor W_f is defined as the function of the current value for the factor V_{f,n_i}. More specifically, it can be determined using the average, standard deviation, maximum and minimum of the values of that factor among the server nodes. This function can be expressed as follows.

$$W_f = f\left(\mu_f, \rho_f, \max_{nodes}(V_{f,n_i}), \min_{nodes}(V_{f,n_i})\right)$$

where

$$\begin{aligned}\mu_f &= \text{The average of } V_{f,n_i} \text{ for all nodes } n_i\\ \rho_f &= \text{The standard deviation of } V_{f,n_i} \text{ for all nodes } n_i\\ \max_{nodes}(V_{f,n_i}) &= \text{The maximum value of factor } p \text{ among all nodes}\\ \min_{nodes}(V_{f,n_i}) &= \text{The minimum value of factor } p \text{ among all nodes}\end{aligned}$$

There can be many different definitions of the function $f()$. In this paper, we use a very simple form using three factors and the standard deviation of the measured values.

$$\mathit{FactorSet} = \{\text{CPU usage(CPU)},\ \text{Network input(Rx)},\ \text{Network output(Tx)}\}$$

$$L_{n_i} = \sum_{f \in FactorSet} W_f V_{f,n_i}$$

$$W_f = \begin{cases} 0.9 & \text{if } \rho_f = \max\limits_{g \in FactorSet}(\rho_g) \\ 0.05 & \text{otherwise} \end{cases}$$

In the equation above, if the standard deviation of the values measured for a factor among all the nodes is greater than those for any other factors, the weight of that factor is set to 0.9. Otherwise, the weight is set to 0.05. For example, if the standard deviations of CPU usage, network input and network output are 20, 10 and 15, respectively, the weights for those factors are set such that $W_{CPU} = 0.9$, $W_{Rx} = 0.05$, $W_{Tx} = 0.05$, respectively.

4 Experiments

4.1 Environments

We tested our algorithm using LVS. Server cluster was constructed as follows:

- Load Balancer × 1
- Server Node × 2
- Client × 3
- All nodes are composed by Pentium4 2.8GHz CPU, 512MB RAM and Ethernet 1Gbps NIC. Linux 2.6.10 is used as OS.

We built a client program which sends requests to the server repeatedly. For each request, the server sends a data packet of 100KBytes size to the client. Then we generated additional loads (CGI and network file I/O) in one server node and estimated how our algorithm schedules the user requests. We also built a daemon program that runs on the server nodes. This daemon program gets the current load information by reading /proc/* system files every second.

We measured the response rate of the server as the request rate was increased. Lastly, we compared the following cases:

- When no additional load is added
- Additional load, LVS uses round-robin scheduling.
- Additional load, fixed weights are given ($W_{cpu} = 1.0$,$W_{rx} = 0.0$,$W_{tx} = 0.0$)
- Additional load, fixed weights are given ($W_{cpu} = 0.5$,$W_{rx} = 0.0$,$W_{tx} = 0.5$)
- Additional load, fixed weights are given ($W_{cpu} = 0.0$,$W_{rx} = 0.0$,$W_{tx} = 1.0$)
- Additional load, adaptive weight assignment is used

4.2 Results

Figure 2 shows the response rate when CGI was performed as additional load (CPU-bound service). It can be seen that the response rate for adaptive weights is not better but similar to those for the case when the weights for CPU is 1.0. Figure 3 shows the response rate when network file I/O was performed as additional load (I/O-bound service). In this case, the response rate for adaptive weights is better than those for any other cases of fixed weights.

Comparing fig. 2 and fig. 3, we can see that the best choice of fixed weights for CPU-bound service (cpu:1 rx:0 tx:0) results in the worst performance for I/O-bound service, and vice versa. However, the adaptive weights assignment shows good performance for both cases.

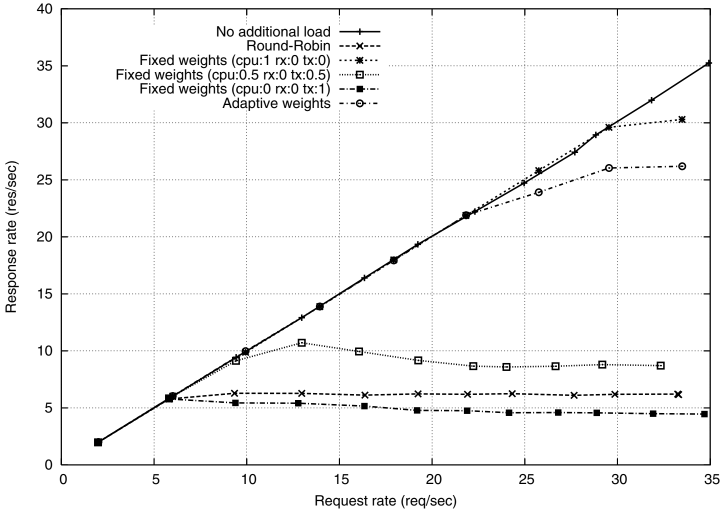


Fig. 2. Response rate for CPU-bound service

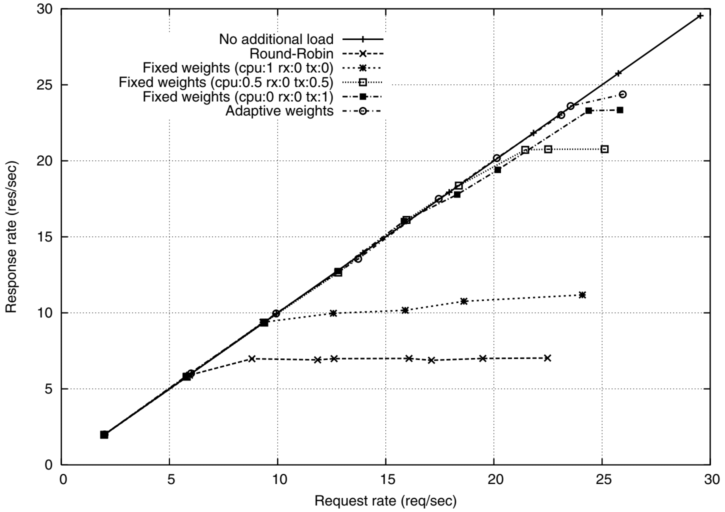


Fig. 3. Response rate for I/O-bound service

5 Conclusion and Future Works

Load balancing is one of the most important factors that determine the performance of a server cluster system. In this paper, we proposed an adaptive and efficient load balancing algorithm for server cluster. In our algorithm, the load balancer measures the loads of server nodes and chooses the node that will process the user request. The load of a server node consists of several factors such as CPU usage and the amount of I/O, and so on. Major factor that causes a bottleneck depends on the characteristic of the service. Our algorithm can identify the factors that cause such bottlenecks and adjust the weight of those factors. The weighted factors are used to compute the load of each server node Consequently, the server can show high performance even when there are changes in the characteristic of the service.

References

1. Wensong Zhang. Linux virtual server for scalable network services. In *Proceedings of the Ottawa Linux Symposium 2000*, July 2000.
2. The linux virtual server project - linux server cluster for load balancing http://www.linuxvirtualserver.org/.
3. H.-K. Baik G.-H. Kim M.-S Park Y.-H. Shin, S.-H. Lee. Agent-based sophisticated and scalable scheduling method. In *Proc. 3rd International Network Conference*, 2002.
4. Lee Jangho Myong-Soon Park Kyeongmo Kang, Sookheon Lee. Assum:agent-based sophisticated scheduling mechanism by using memory utilization in web server cluster. 2003.
5. Valeria Cardellini, Michele Colajanni, and Philip S. Yu. Dynamic load balancing on web-server systems. *IEEE Internet Computing*, 3(3):28–39, 1999.
6. Chad Yoshikawa, Brent Chun, Paul Eastham, Amin Vahdat, Thomas Anderson, and David Culler. Using smart clients to build scalable services. In *Proceedings of the USENIX 1997 Annual Technical Conference*. USENIX Association, 1997.
7. Thomas T. Kwan, Robert McCrath, and Daniel A. Reed. NCSA's world wide web server: Design and performance. *IEEE Computer*, 28(11):68–74, 1995.
8. M. Colajanni, P. S. Yu, and D. M. Dias. Analysis of task assignment policies in scalable distributed Web-server systems. *IEEE Transactions on Parallel and Distributed Systems*, 9(6):585–598, 1998.

Design and Performance Analysis of a Message Scheduling Scheme for WLAN-Based Cluster Computing*

Junghoon Lee[1], Mikyung Kang[1], Euiyoung Kang[2], Gyungleen Park[1],
Hanil Kim[2], Cheolmin Kim[2], Seongbaeg Kim[2], and Jiman Hong[3],**

[1] Dept. of Computer Science and Statistics, Cheju National Univ
[2] Dept. of Computer Education, Cheju National Univ
[3] School of Computer Science and Engineering, Kwangwoon Univ.,
690-756, Jeju Do, Republic of Korea
{jhlee, mkkang, glpark, hikim, cmkim, webkey, sbkim}@cheju.ac.kr,
gman@daisy.kw.ac.kr

Abstract. This paper proposes and analyzes the performance of a task scheduling scheme that couples network schedule for master-slave style parallel computing cluster built on top of wireless local area networks. When a transmission fails, the proposed scheme selects another data and destination that can replace the subtask scheduled on the unreachable node with minimal cost, rather than hopelessly retransmits on the bad channel. Simulation results performed via ns-2 event scheduler, show that the proposed scheme minimizes the task migration, improves the computation time by maximally 35.4 %, increases the probability of successful retransmission, and finally survives the network failure as long as at least one node is reachable at each instance of time.

1 Introduction

Networks of standard workstations provide an attractive scalability in terms of computation power and low cost. It is becoming strongly competitive compared with expensive parallel machines[1]. Cluster as well as grid computing is a new computation idea that takes advantage of independent and different resources to build an unified resource for massive computation, distributed computation, data storing, and so on. While the fixed networks of computers constitute the lowest cost as well as the most available parallel computer, the proliferation of wireless devices such as PDA, telematics, and so on, allows to expand the parallel virtual machine. Recent advances in wireless communication technology are making WLAN (Wireless Local Area Network) an appealing transmission media for parallel and distributed computing on networked computers[2].

* This research was supported by the MIC, Korea, under the ITRC support program supervised by the IITA.

** Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 558–566, 2006.

The use of MPI (Message Passing Interface), PVM (Parallel Virtual Machine), or other variants can work in wireless environment because they are built on top of TCP/IP and therefore the physical medium does not impose any restriction[3]. However, it is not clear how well this mechanism fits for wireless networks, since wireless channels are subject to unpredictable *location-dependent* and *bursty* errors. A parallel application may fail altogether if the wireless connection stays down too long. This problem fatally affects MPI parallel programs because the default behavior in case of network failure is the immediate termination of application. Though such a problem can be somehow relieved using a dynamic process migration functionality of MPI or other similar mechanisms, corresponding overhead and waste of bandwidth are significant, as an intermediate processing is discarded and a new task is created at some other node.

To optimize the computing speed, the number of such migrations should be kept as small as possible. However, considering bursty and unpredictable nature of wireless channel errors, immediate retransmission does not seem to be appropriate, as the subsequent retransmissions to the original destination may fail repeatedly. To solve this problem, this paper obviates hopeless retransmissions based on the main idea that the undelivered message is not necessarily retransmitted to the current destination. Namely, when a node becomes unreachable, the master should decide a new node that can take the task originally assigned to the unreachable node. Consider a process that needs two parameters, A and B, and node 0 has A while node 1 has B, respectively. If master fails to transmit B to node 0, due to bad channel condition, it is better to try to send A to node 1 rather than retransmit B to node 0.

The rest of this paper is organized as follows: Section 2 describes the background of this paper, including related works on cluster computing on WLAN as well as IEEE 802.11 WLAN standard itself. Then Section 3 proposes a message scheduling scheme on WLAN. After demonstrating the performance measurement results obtained via simulation using ns-2 in Section 4, Section 5 finally concludes this paper with a brief summarization and the description of future works.

2 Background

2.1 Related Works

MPI provides a fault tolerant mechanism that can cope with dynamic changes resulted from network errors. This scheme consists of spawning slave processes one by one as new portable or fixed nodes become available and creating an independent intercommunicator for each master and slave process. Additionally, FT-MPI is a fault-tolerant MPI implementation that includes some dynamic process management functionalities[4]. It lets the communicators to be in an intermediate state and they can be rebuilt so the application can recover from a fail. FT-MPI survives the crash of $(n-1)$ processes in a n-process job, and, if required, can respawn/restart them. However, it is still the responsibility of the application to recover the data-structures and the data on the crashed processes.

LAMGAC is at first a library that makes it easy to program a MPI parallel application in a LAN-WLAN cluster where the parallel virtual machine can change during process execution[5]. LAMGAC is extended to detect temporary or permanent disconnections of the wireless channel by implementing a new function named as *LAMGAC_Fault_detection*[3]. This function is invoked by the master node whenever it needs to check if there is a physical connection to one slave process and therefore guarantee a successful message interchange between them. Standard *ping* application enables the library to determine which slave node is reachable and then return this information back to the server. The master executes the attachment and detachment protocol and master can spawn slave processes creating a new intercommunicator for each spawning.

Legion considers wireless and mobile devices as significant computation resources for the Grid[1]. Legion's object architecture makes the system capable of dealing with the high degree of intermittent connectivity associated with mobile and wireless devices. It implemented a dynamic *Resource Adaptation Layer* that will query and modify the characteristics of the system according to changing user and device needs. This function supports the ability to select a minimal subset, which is crucial for such small-memory mobile and wireless devices. However, the above mechanisms are based on the end-to-end error recovery as well as process migration.

2.2 IEEE 802.11 WLAN

The IEEE 802.11 was developed as a MAC standard for WLAN[6], and we exploited a computing architecture of LAMGAC as shown in Fig. 1. It considers master-slave parallel application in which for every iteration the processes must synchronize to interchange data. The master is in the access node (AN) and the slaves are in the member node (MN). MN can be mobile or fixed node connected via WLAN. Based on the infrastructure WLAN where each MN only communicates with AN, the master distributes the task. However, even in the *ad hoc* mode, we can designate a node to play a role of AN to schedule the transmission. For downlink channel (AN to MN), AN schedules those packets that are generated at AN or arrived from another cluster, while for uplink channel (MN to AN), AN sequentially polls each node according to a specific schedule[7].

The 802.11 radio channel is modeled as a Gilbert channel[8]. In this model, p denotes the transition probability from state *good* to state *bad* while q the probability from state *bad* to state *good*[8]. The pair of p and q represents a range of

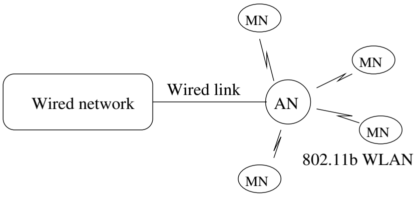


Fig. 1. WLAN-based computing architecture

channel conditions, and it has been typically obtained by using the trace-based channel estimation. The average error probability, denoted by ϵ, and the average length of a burst of errors are derived as $\frac{p}{p+q}$ and $\frac{1}{q}$, respectively. The packet is received correctly if the channel is in state *good* for the whole duration of packet transmission, otherwise, it is received in error. In addition, according to WLAN standard, poll, transmission, and acknowledgment are atomic, namely, these steps must complete in their entirety to be successful. Senders expect acknowledgment for each transmitted frame and are responsible for retrying the transmission. After all, error detection and recovery is up to the sender station, as positive acknowledgments are the only indication of success. If an acknowledgment is expected but does not arrive, the sender considers the transmission failed.

3 Network Schedule for Cluster Computing

3.1 Channel Estimation

We take the estimation method from Bottiglieno's work[9]. To trace the channel status, AN maintains a state machine associated with each member node. The state can be either *good* or *bad* according to the channel condition. The channel condition is estimated as follows: The ACK/NAK is sent from the receiver to AN as soon as it receives a packet. If the AN does not receive an ACK/NAK within predefined time-out interval, the packet will be assumed to be lost. Then the state triggers to *bad*. AN sets the state to *good* whenever it receives from the corresponding node, namely, a MAC-layer acknowledgment in response to a data frame, a CTS frame in response to a RTS frame, or any other error-free frame. The AN sets the state to *bad* after a transmission failure. Each *bad* channel has its own counter, and when a counter expires, the AN attempts to send a single data frame to check the channel status. The duration of timer is reset to its initial value upon a transmission from *bad* to *good*, and the value is doubled whenever the probing fails in *bad* state. The value of timer should be set small so as to quickly recover from short channel error period. This probing procedure is carried out via currently idle link, either uplink or downlink.

3.2 Description of Basic Idea

The ultimate goal is to guarantee the successful completion of the parallel program even in the presence of wireless link failures. We use a WLAN as a underlying network, where master process runs on AN, while slaves on member nodes. A task is typically divided into several subtasks, each of them is assigned to other slave process, and then the partial results are integrated to produce a final result. The master process is in charge of the dynamic data distribution to the remainder processes of the parallel program. In this procedure, transmission of argument and result impacts the efficiency of parallel application. The node set may change dynamically in runtime as a new node can be added or an existing node can be detached from the working group. The network schedule

Command	Arg	Subtask
DoComp	—	
Put&Comp		
Put		—

Fig. 2. Command set

totally depends on AN which polls for upstream schedule and decides the transmission order for downstream[10]. Each node only communicates with AN and its channel is in one of two states, namely, *good* state or *bad* state.

All slaves have the program code for the given task as in the MPI programming model, and master node can initiate their executions by downloading relevant parameters via downlink channel. Slaves iteratively wait for the command from the master on the downlink channel and perform the command. After the completion of subtask, the slave returns the result to master via uplink channel. AN schedules the transmission of uplink channel by a polling mechanism. Fig. 2 shows the command set defined for the proposed computing scheme. The command set consists of *DoComp*, *Put&Comp*, and *Put*, and each command includes *Arg* and *Subtask* fields. *Arg* can include the necessary parameter to perform a specified mission in *Subtask* field. *DoComp* is used when a node already has enough parameters to compute a specific subtask, thus it only specifies subtask without any argument. *Put&Comp* makes to perform a subtask and with the parameter enclosed in the command as well as the one it already holds. *Put* is the command that is used when every available node has insufficient data to complete a subtask. Even with the parameter enclosed in this command, the receiver does not have enough data to fulfill any subtask. Thus *Subtask* field contains nothing. Generally, *Put&Comp* follows the *Put* command.

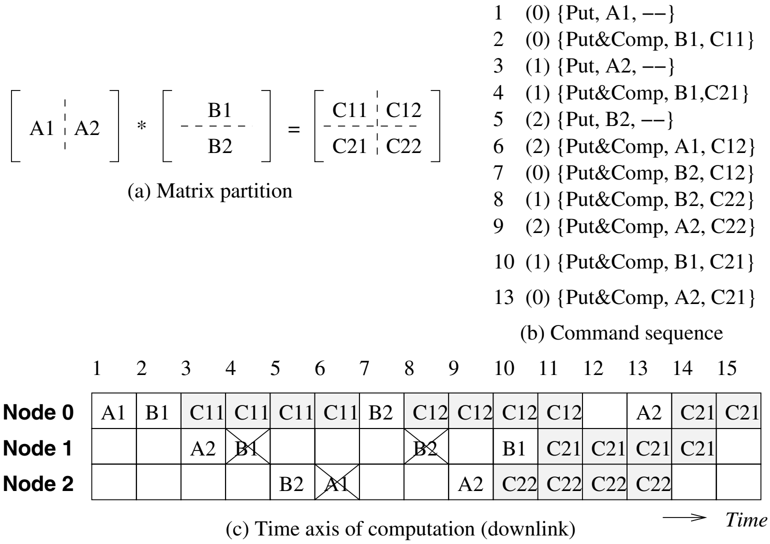
As the master works on AN, the network schedule is decided from the computation schedule, while the computation schedule takes into account the information including whether a node is currently busy or not and whether its channel is in good state or bad state. With these data, the *task scheduler* on AN, decides the command sequence and delivery schedule as follows:

For a node in good channel status and not computing,
Rule 1. If N_i has enough data to perform a subtask S_j, send to N_i {DoComp, –, S_j }. If transmission succeeds, mark S_j as *InProgress* and N_i as *Computing* S_j. If there is no candidate, go to step 2.
Rule 2. If N_i can perform a subtask, S_j, which is not started, with a data A_k, send to N_i {Put&Comp, A_k, S_j}. On successful transmission, mark S_j as *InProgress* and N_i as computing S_j and also mark N_i has A_k. If there is no candidate, go to step 3.
Rule 3. Select the parameter that is required to proceed the subtask in *NotStarted*. If transmission succeeds, mark N_i has A_k.

In addition to such rules, task scheduler performs following management functions. Namely, It checks if all subtasks are completed. If so, the scheduler finalizes the job and merges the results. In addition, if the status of a node marked as executing S_j triggers to *bad*, save the current status and set the status as $NotStarted$. When the channel gets back to *good* status, restore the saved status, that is, the subtask it was executing. Notice that even if a node temporarily disconnected, its computation goes on.

3.3 Example

We will show a transmission scenario generated by the proposed task scheduling scheme for the typical matrix multiplication with 3 nodes. As shown in the Fig. 3(a), the problem is divided into 4 multiplications of submatrices, namely, A1·B1, A1·B2, A2·B1, and A2·B2. For simplicity, we assume that each submatrix is as large as a slot time that corresponds to the transmission of a single packet. In addition, the computation time is an integer multiple of slot time, say 4 slots. Task scheduler decides the submatrix to send at each start of slot. Notice that the *DoComp* command does not need argument, so the transmission time is small enough to be ignored.



$$\begin{bmatrix} A1 & A2 \end{bmatrix} * \begin{bmatrix} B1 \\ B2 \end{bmatrix} = \begin{bmatrix} C11 & C12 \\ C21 & C22 \end{bmatrix}$$

(a) Matrix partition

1 (0) {Put, A1, −−}
2 (0) {Put&Comp, B1, C11}
3 (1) {Put, A2, −−}
4 (1) {Put&Comp, B1,C21}
5 (2) {Put, B2, −−}
6 (2) {Put&Comp, A1, C12}
7 (0) {Put&Comp, B2, C12}
8 (1) {Put&Comp, B2, C22}
9 (2) {Put&Comp, A2, C22}
10 (1) {Put&Comp, B1, C21}
13 (0) {Put&Comp, A2, C21}

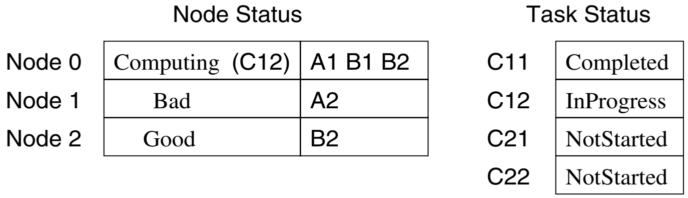
(b) Command sequence

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Node 0	A1	B1	C11	C11	C11	C11	B2	C12	C12	C12	C12		A2	C21	C21
Node 1			A2	~~B1~~				~~B2~~		B1	C21	C21	C21	C21	
Node 2					B2	~~A1~~			A2	C22	C22	C22	C22		

(c) Time axis of computation (downlink)

Fig. 3. Example of network scheduling

In Fig. 3(b), at time 1, all nodes do not have any argument, but also are not computing. The task scheduler chooses to send A1 to node 0 according to Rule 3. At time 2, by Rule 2, the scheduler selects {Put&Comp, B1, C11} as node 0 can compute C11 if it gets B1. From time 4, node 2 triggers to bad channel, so the transmission of {Put&Comp, B1, C21} fails. The transmission to node 2 at time 6 also fails due to the same reason. At time 8, node 1 is selected to compute



Node Status

Node 0	Computing (C12)	A1 B1 B2
Node 1	Bad	A2
Node 2	Good	B2

Task Status

C11	Completed
C12	InProgress
C21	NotStarted
C22	NotStarted

Fig. 4. Snapshot of data structure at time 9.0

C22 after receiving B2, however, the transmission fails. C22 can be calculated at node 2 with A2, so the corresponding command is sent to node 2, without trying to retransmit B2 to node 1 in contrast to MPI approach.

Fig. 4 shows the snapshot of data structure at time 9, where the only candidate is Node 2, as Node 0 and 1 are computing and in bad status, respectively. By Rule 2, as C22 is not yet started and Node 2 has B2, the task scheduler selects action as {Put&Comp, A2, C22}. Finally, at time 13, the scheduler redundantly send A2 to node 0 to compute C21 that is also being computed in node 1. This is because node 1 goes to *bad* channel status at time 13, and thus the task status of C21 changes from *InProgress* to *NotStarted*. If the channel recovers, the result is safely sent to the master process, finalizing the whole computation procedure. Otherwise, newly started calculation replaces the original one.

4 Performance Analysis

The simulation is performed using ns-2 event scheduler that enables to construct a flexible event-driven simulation environment[11]. In this experiment, each member node is assumed to have equal computing power, as we are mainly concerned on the efficiency of network schedule for parallel computing in wireless channel. The simulation revisits the matrix multiplication for the target parallel application. The performance measurement compares the execution time of our scheme with that of the traditional MPI according to the number of nodes and channel error rate, respectively. Both schemes divide the given tasks into A1·B1, A1·B2, A2·B1, and A2·B2, each of them is performed at each node. For each experiment, the simulation runs 20 times and then the execution times are averaged to achieve the final result.

Fig. 5 plots the execution time according to the number of nodes. In this experiment, the message error rate is set to 0.25 and the computation time is 3 slots. When the number of nodes is equal to or greater than 6, the performance of parallel computing remains constant, since the computing resources are sufficiently available. This situation is same on the traditional scheme. On small number of nodes, the effect of channel error increases, as the task scheduler cannot find the appropriate action when all channels stay down or are already computing assigned task. The figure also plots the execution time of ideal case when all channels stay good during the entire computation process. As the computation time is 3 slots, the completion times of error-free case are all same when there are 3 or more nodes. Finally, the gap between the proposed scheme and the error-free case results from the overhead due to transmission failure.

Fig. 6 exhibits the execution time according to the message error rate. The error rate is denoted with parameter p and q, and if the ratio is 3.0, the message error rate is $\frac{1}{1+3} = 0.25$. The number of nodes is 4 while the computation time is 3 slots. As expected, the performance gap between the proposed scheme and the traditional scheme is maximized by 35.4 % when the channel is highly unstable. We observed that if the message error rate is less than 10%, that is, p to q ratio is more than 9.0, almost all packet transmissions succeed, so there is only a little performance enhancement compared with the traditional computing scheme. The message error rate is the function of bit error rate and the message length, and the rate is chosen to be rather high to magnify the robustness of proposed scheme. For both schemes, when the computing time is high, the effect of message loss due to disconnection is more critical, as another node should take the subtask on the disconnected node. However, as the frequency of disconnection is not so high, the effect of disconnection is flattened by averaging the mass of experiment results.

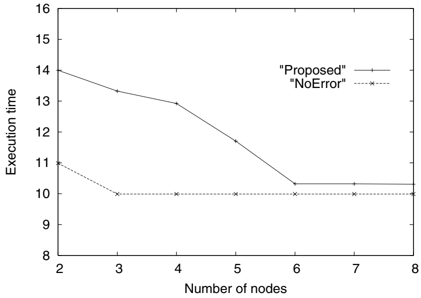


Fig. 5. # of nodes vs. execution time

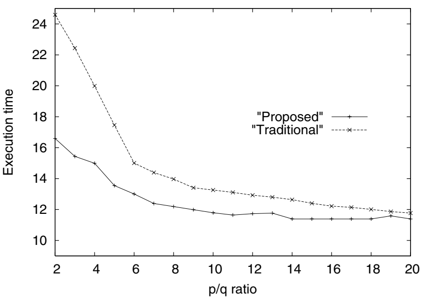


Fig. 6. Error rate vs. execution time

5 Conclusion

In this paper, we have proposed and analyzed the performance of a task scheduling scheme that couples network schedule for master-slave style parallel computing cluster built on top of wireless local area networks. When a transmission fails due to temporary network disturbance or long time disconnection, the proposed scheme does not try to retransmit the erroneously transmitted packet to the unreachable destination, but it selects another data and destination to perform the subtask scheduled on the currently unreachable destination node. With this idea, we developed a heuristic method for parallel applications on the unreliable wireless network and applied to the traditional matrix multiplication problem. As the proposed scheme can minimize or eliminate the task migration that imposes significant overhead and also improve the probability of successful retransmission, it outperforms the traditional MPI style framework. The simulation result performed by ns-2 event scheduler shows that the proposed scheme can improve the computation speed for the whole range of channel error rate and computation time. Most importantly, the experiment also demonstrates that the proposed scheme reliably performs the parallel application in the sense that it can survive the network failure as long as at least one node is reachable at each instance of time.

For an application that needs more complex topology, the cluster-based communication model can be exploited[12]. As a future work, we are to apply the proposed heuristic to various applications on such wireless clusters. In addition, we believe that load balancing issues should be reinforced to our message scheduling scheme to make the WLAN cluster more practical, as the general cluster consists of heterogeneous mobile or fixed node, that is, processors have different speeds, memory resources, variable external load, and even the different network interface speed.

References

1. Clarke, B., Humphrey, M.: Beyond the Device as Portal: Meeting the Requirements of Wireless and Mobile Devices in the Legion Grid Computing System. 2nd International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing (2002)
2. McKnight, L., Howison, J., Bradner, S.: Wireless grids: Distributed resource sharing by mobile, nomadic, and fixed devices. IEEE Internet Computing (2004) 24-31
3. Macías, E., Suárez, A.: Solving Engineering Applications with LAMGAC over MPI-2. 9th EuroPVMMPI International Conference (2002)
4. Fagg, G., Bukovsky, A., Dongarra, J.: Fault Tolerant MPI for the HARNESS Meta-Computing System. Lecture Notes in Computer Science, Vol. 2073. Springer-Verlag, Berlin Heidelberg New York (2001) 355-366
5. Macías, E., Suárez, A.: A Mechanism to Detect Wireless Network Failures for MPI Programs. 4th DAPSYS International Conference (2002)
6. IEEE 802.11-1999: Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. also available at *http://standards.ieee.org/-getieee802* (1999)
7. Choi, S., Shin, K.: A unified wireless LAN architecture for real-time and non-real-time communication services. IEEE/ACM Trans. on Networking (2000) 44-59
8. Bai, H., Atiquzzaman, M.: Error modeling schemes for fading channels in wireless communications: A survey. IEEE Communications Surveys, Vol. 5, No. 2 (2003) 2-9
9. Bottigliengo, M., Casetti, C., Chiaserini, C., Meo, M.: Short term fairness for TCP flows in 802.11b WLANs. Proc. IEEE INFOCOM (2004)
10. Lee, J., Kang, M., Jin, Y., Kim, H., Kim, J.: An efficient bandwidth management scheme for a hard real-time fuzzy control system based on the wireless LAN. Lecture Notes in Artificial Intelligence, Vol. 3642. Springer-Verlag, Berlin Heidelberg New York (2005) 644-659
11. Fall, K., Varadhan, K.: Ns notes and documentation. Technical Report, VINT project, UC-Berkeley and LBNL (1997)
12. Liu, K., Li, J.: Mobile Cluster Protocol in Wireless Ad Hoc Networks. Proceedings of International Conference on Communication Technology (2000)

A Method for Efficient Malicious Code Detection Based on Conceptual Similarity

Sungsuk Kim[1], Chang Choi[1], Junho Choi[2,*], Pankoo Kim[2,*], and Hanil Kim[3]

[1] Dept. of Computer Science, Chosun University, Gwangju 501-759 Korea
sezeroot@empal.com, enduranceaura@gmail.com
[2] Dept. of Computer Engineering, Chosun University, Gwangju 501-759 Korea
{spica, pkkim}@chosun.ac.kr
[3] Dept. of Computer Education, Cheju National University, Cheju 690-756 Korea
hikim@cheju.ac.kr

Abstract. Nowadays, a lot of techniques have been applied for the detection of malicious behavior. However, the current techniques taken into practice are facing with the challenge of much variations of the original malicious behavior, and it is impossible to respond the new forms of behavior appropriately and timely. With the questions above, we suggest a new method here to improve the current situation. Basically, we use conceptual graph to define malicious behavior, and then we are able to compare the similarity relations of the malicious behavior by testing the formalized values which generated by the predefined graphs in the code. In this paper, we show how to make a conceptual graph and propose an efficient method for similarity measure to discern the malicious behavior. As a result of our experiment, we can get more efficient detection rate. It can be used in detecting malicious codes in the script based programming environment of many kinds of embedded systems or telematics systems.

1 Introduction

With increasing Internet use, the number of malicious codes is spreading rapidly through Internet services. The most common type of malicious code is viruses. But malicious codes are evolving more complex and smart. Malicious codes are designed to affect your computer adversely such as making your computer to malfunction, disseminate information, or distributed service attack. Especially, malicious codes prepared with VBScript based on the windows operating system are on the rise. In order to detect these VBScript malicious codes, signature based scanning is used most commonly. It can not detect the malicious codes not known or source codes with their structure modified[7]. In order to detect unknown malicious codes, a new approach is needed conceptually differently from existing methods.

Thus, we propose a new method of detecting malicious codes using conceptual graphs while decreasing error rates compared with the existing methods. For this purpose, we define the concept and relation of source codes and propose a method of evaluating similarity of malicious codes after creating a conceptual graph using the concept and relation.

[*] Corresponding authors.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 567 – 576, 2006.

2 Related Works

2.1 Existing Methods of Detecting Malicious Codes

Signature based scanning is used most extensively to detect malicious codes. With this method, a specific string is detected in the malicious code to diagnose maliciousness. It is speedy and differentiate malicious codes accurately; however, it is helpless when it comes to new malicious codes. Heuristic analysis can be divided into static heuristic analysis through code analysis and dynamic heuristic analysis through virtual emulation. Static heuristic analysis is a method in which a malicious code is defined when abnormal return values are detected in the method or internal function. This method is relatively fast and offers a high detection rate; however, the false positive rate can be occurred in detecting a normal script as a malicious code. Dynamic heuristic analysis defines malicious codes by system calling through virtual emulation and detecting changes in resources. With this method, accurate diagnosis is possible when the malicious code actually functions, but virtual emulation diagnosing malicious codes is difficult to simulate and requires much time[1,2,7].

2.2 Conceptual Graphs and CGIF

A conceptual graph can be defined as a knowledge representation language integrating various semantic networks that is logical, concise, expressive to the point of a natural language using conceptual diagrams, easily understood by people, and can be easily used by computer for natural language treatment[4,5]. For example, Figure 1 shows the conceptual graph of the sentence, “John is going to Boston by Bus".

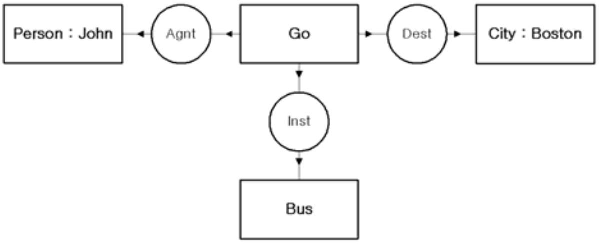


Fig. 1. Conceptual Graphs

The rectangles in Figure 1 represent the concepts; the circles, the relationship between the concepts. The direction arrows connect each note. "Agnt", "Dest", and "Inst" represent relationship; and "John", "Boston", and "Bus" are concepts. The expression "Person : John" represents the concept instance of "John" as a "person". Furthermore, a conceptual graph can be converted into the expanded BNF expression, CGIF (Conceptual Graph Interchange Format). Table 1 is the CGIF of the conceptual graph in Figure 1.

Table 1. CGIF expression of conceptual graph

01 :	[City*a:‘Boston’]	05 :	(agent?d?c)
02 :	[Bus*b:‘’]	06 :	(dest?d?a)
03 :	[Person*c:‘John’]	07 :	(inst?d?b)
04 :	[Going*d:‘’]	08 :	

Similarity among the concepts can be evaluated in the conceptual graphs expressed so that significant similarity can be evaluated and compared among the conceptual graphs. Thus, the method of detecting malicious codes is proposed in this paper by evaluating conceptual graph expression for script source codes and concept similarity.

3 Conceptual Graph Expression in Malicious Codes

3.1 Definition of VBScript Concept and Relation

In this section, the sentence structure and vocabulary of VBScript are analyzed to define the method of conceptualization and relationship. For the conceptual expression of VBScript, we first need to define the concept of source code and relationship. For the definition of concept and relation in programming source codes such as VBScript, the elements composing the programming language are classified. Each component forming the structure of each stratum is defined usually.

Table 2. Definition of VBScript source code concept (example)

Concept		Exposition	Related grammar
Procedure		A serious of sequence and process to resolve a problem	Sub, End Sub
State-ment	Conditional	The sentence that could control program execution into different directions according to the given condition.	If…Then..else, Select Case
	Loop	The program sentence that could execute a serious of commands repeatedly.	Do…Loop, While…Wend, For…Next etc.
	Error	Execution into a different method rather than executing into the expected method of a certain operation.	On Error Resume Next, Error
Operator	Compari-son	Comparing the size of two inputted data transmitted	‘<’, ‘=’, ‘< >’
	…..		

Based on the structure of strata classified, the concept is defined as Table 2 using MSDN (Microsoft Developer Network Library) and VBScript Language Reference. Furthermore, the relationship within the concept in source codes is defined in Table 3. For example, the grammatical concept of "procedure" is related with {Condition, Argument}[17].

Table 3. Definition of VBScript source code relation

Relation	Definition	Relation conditions	
		Upper concepts	**Lower concepts**
Condition	Conditions to distinguish distribution	Conditional, Loop	Statements, Operator, Assign, Procedure(-Call), String, Variable
Contains	Concepts including other concepts	*	*
Comment	Notes	*	String
Return	The concept returning the returning value	Function, Method	Function, String, Variable

(* : A set of all concepts and concepts including relation conditions)

3.2 Definition of Malicious Code Pattern and Expression of Conceptual Graph

In this section, the conceptual graph is expressed according to the relationship with the defined concept to extract the malicious code statement according to malicious activities. The execution of malicious code described in VBScript generates an object, which is completed by calling the method. The following is Window's objects related with malicious code.

- Scripting.FileSystemObject
- WScript.Network
- WScript.Shell
- Outlook.Application

Among these, the "Outlook.Application" exists only in the system installed with Outlook and the rest, in the system installed with WSH (Windows Script Host). The typical malicious code, "Love Letter", manipulates Windows registry, copies through E-mail, eliminates certain files, and produces malicious HTML files. In order to analyze the codes executing malicious behavior, after expressing the concept and relationship produced on the malicious code with a conceptual graph, a formal transformation is needed for system application. Table 4 shows a list of self-duplicating malicious codes in a system folder. Referring to this sample, the general pattern of malicious codes is expressed in a conceptual graph. Figure 2 is the standardized conceptual graph. The concepts of "statements", "method", and "arguments" included in this method are classified. Using their relationship, Figure 2 is the conceptual graphs of self-duplicating malicious codes. This shows the procedure of a malicious code self-duplicating in the local system expressed in a conceptual graph. In order to carry out a malicious behavior, the concept of "statement" (area A) and the concept of "method" (area B) duplicating the statement are included. The concept to be used in the malicious activity is generated through "Statements : Set". Using this concept, the site of malicious code and the site of duplication are determined using the concept of "Method : CopyFile". Using the above method, once malicious codes presented in various forms are expressed in conceptual graph, even when the malicious codes with a modified source and new malicious code is generated, the malicious activity can be recognized conceptually.

Table 4. The patterns of "self-duplicating" malicious code(example)

malicious code Sample 1	On Error Resume Next *Set Obj_A = Createobject("scripting.filesystemobject")* Obj_A.*copyfile wscript.scriptfullname*, Obj_A.*GetSpecialFolder(0)*& -"\xxx.jpg.vbs"
malicious code Sample 2	main = "c:\www.symantec.com.vbs" *Set maincopy = CreateObject("Scripting.FileSystemObject")* maincopy.*CopyFile WScript.ScriptFullName*, main
.....	.....

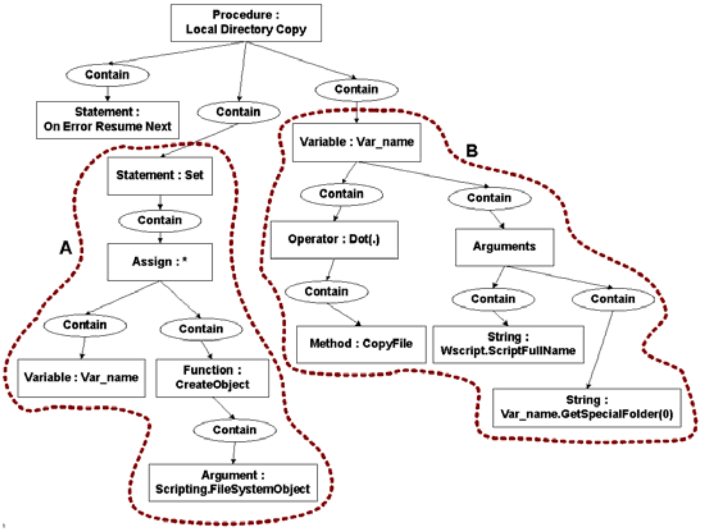


Fig. 2. A conceptual graph of a "self-duplicating" malicious code

4 Evaluating Similarity for the Detection of Malicious Codes

By evaluating similarity, the presence of malicious codes and the risk of malicious codes could be measured in a conceptual graph. In order to accurately evaluate malicious behavior, when defining the concepts and relationship, it is desirable to grant the degree of risk on malicious behavior. Thus, a value was placed on the grammatically elements of conceptual graphs considering the characteristics of malicious codes in this paper. When similarity is evaluating not considering the value according to the characteristics of the malicious code, the risk of recognizing a normal code as a malicious code is high. Furthermore, emphasis is placed on the grammatically elements used frequently in a VBScript code considered as a malicious code to evaluate similarity.

In order to obtain an equation for evaluating similarity, 3 steps are needed. The first step is to measure the value on the concept types of referent composing the conceptual graph. The second step is to induce similarity in the conceptual graph. The third step is to place a value according to the frequency of concepts essential for malicious behavior by applying the optimal values for the defined concepts, referents, and relationship. Then, the values are induced using the following process.

Step 1 : Among the conceptual graph factors of VBScript, the relational equation was defined for the concept type and referent. The concept type and referent were defined using the following symbols.

Definition 1. c_1^t : concept type, c_1^r : referent

One of the concept c_1 , would have the two values, ie., c_1^t , c_1^r . The value of the concept type c_1^t is that granted based on concept importance. The referent c_1^r signifies the node including the concept.

Table 5. Priority based on concept importance

Rank	Name	Rank	Name
1	Procedure	7	Assign
2	Procedure-Call	8	Operator
3	Function	9	Properties
4	Object	10	Arguments
5	Method	11	Variable
6	Statement	12	String

Thus, the c_1 can be expressed as the following Equation 1 based on the concept type and referent.

Equation 1. $c_1 = c_1^t \times c_1^r$

Step 2 : The concept type similarity and referent similarity between the concepts c_1 and c_2 are defined as the following Equation 2 using the concept c proposed in Step 1.

Equation 2. Concept type similarity : $sim(c_1^t, c_2^t) = c_1^t \cdot c_2^t$

Referent similarity : $sim(c_1^r, c_2^r) = c_1^r \cdot c_2^r$

Using the Equation 2, overall similarity between the two graphs G_1, G_2 can be defined as Equation 3.

Equation 3. $sim(G_1, G_2) = \sum_{i=1}^{n} \sum_{j=1}^{m} sim(c_i^t, c_j^t) \cdot sim(c_i^r, c_j^r) \cdot (c_i) \cdot (c_j)$

When the concept union including the concept graph G_1, ie., $\{c_1, c_2, \ldots c_n\}$, G_2, is defined as $\{c_1, c_2, \ldots c_m\}$, the value multiplying each concept factor, ie., concept type similarity and referent similarity, is multiplied by the value of each concept.

Step 3 : Since ordinary conceptual graphs were used on the values obtained in Step 2, to evaluate similarity of malicious codes, a value was placed on the frequently seen concept considering the frequency of the concepts related with the malicious code. Thus, by considering the major relationship among the concepts used in the malicious code, the value of similarity that could be applied in the malicious code is calculated.

Definition 2. $w(r)$: Relation weight

The relative value used when measuring similarity expresses the importance of concept and relationship closely related with the malicious code. According to the relative value, the value obtained from a malicious code can be larger than a normal code even with the same concept relationship. Equation 3 obtained in Step 2, considering the relative value $w(r)$, similarity is measured for the conceptual graphs of malicious code. The final equation for measuring similarity is as Equation 4.

Equation 4.

$$fsim(G_1, G_2) = \sum_{i=1}^{n} \sum_{j=1}^{m} sim(c_i^t, c_j^t) \cdot sim(c_i^r, c_j^r) \cdot w(r_i) \cdot (c_i) \cdot w(r_j) \cdot (c_j)$$

Based on the final equation shown in Equation 4, similarity value can be calculated as follows between the two graphs G$_1$, G$_2$, for example,

fsim(G$_1$, G$_2$)

= sim([Loop:*],[Loop:*])□[Loop:*]□ [Loop:*]□(Contain?a?b)□*(Contain?a?b)*

+ sim([Loop:*],[String:sleep])□[Loop:*]□[String:sleep]□(Contain?a?b)□*(Contain?a?b)*

+ sim([Block:*],[Loop:*])□[Block:*]□[Loop:*]□(Contain?a?b)□(Contain?b?c)□*(Contain?a?b)*

+ sim([Block:*],[String:sleep])□[Block:*]□[String:sleep]□(Contain?a?b)□(Contain?b?c)□*(Contain?a?b)*

+ sim([String:wait],[Loop:*])□[String:wait]□[Loop:*]□(Contain?b?c)□*(Contain?a?b)*

+ sim([String:wait],[String:sleep])□[String:wait]□[String:sleep]□(Contain?b?c)□*(Contain?a?b)* = 0.9437 .

If the two conceptual graph G$_1$, G$_2$ are the same, the value measured would be "1", whereas when there is no similarity, the value would be 0. Furthermore, as shown in the final equation in Step 3, when measurement is done considering the relationship value, similarity of malicious code reflecting the characteristics of the malicious code can be measured.

5 Experiments and Evaluation

In this section, the results of the experiment done to evaluate similarity in malicious code using conceptual graphs are shown. For the experiment, a total number of 130 VBScript malicious code samples and 20 regular VBScript samples were used. The samples for malicious code were those from sources distributed in the Internet, collected in-house, and prepared using the malicious code generator, "VBS Worm

Generator" program. For the experiment, the samples of malicious codes referring to address and executing mail sending using the MS Windows's "Outlook.Application" object were placed in "A Group", those including codes similar to malicious code "B Group", and those normal codes not containing malicious codes and only referring to address were placed in "C Group".

Table 6. 'E-mail sending' malicious code CGIF

01 :	[String*a:‘0’]	24 :	[String*x:‘i’]
02 :	[Statement*b:‘Set’]	25 :	[Statement*y:‘Set’]
03 :	[Variable*c:‘NoteItem’]	26 :	[Varialbe*z:‘ObjApp’]
04 :	[Object*d:‘ObjApp’]	27 :	[String*aa:‘Outlook’]
05 :	[ReferenceOP*e:‘.’]	28 :	[Function*ab:‘CreateObject’]
06 :	[Variable*f:‘AddrList’]	29 :	[ReferenceOP*ac:‘.’]
07 :	[Object*g:‘AddressLists’]	30 :	[Object*ad:‘NoteItem’]
08 :	[Statement*h:‘Set’]	31 :	[Method*ae:‘Add’]
09 :	[Object*i:‘ObjNS’]	32 :	[Object*af:‘Attachm’]
10 :	[Object*j:‘NoteItem’]	33 :	[ReferenceOP*ag:‘.’]
11 :	[Object*k:‘AddressEntries’]	34 :	[Method*ah:‘Send’]
12 :	[Variable*l:‘ObjNS’]	35 :	[Object*ai:‘NoteItem’]
13 :	[Variable*m:‘CurrentAddr’]	36 :	[Property*aj:‘Attachments’]
14 :	[Object*n:‘Addr’]	37 :	[Variable*ak:‘Attachm’]
15 :	[Statment*o:‘Set’]	38 :	[Statement*al:‘Set’]
16 :	[ReferenceOP*p:‘.’]	39 :	[ReferenceOP*am:‘.’]
17 :	[Properties*q:‘To’]	40 :	[Method*an:‘CreateItem’]
18 :	[ReferenceOP*r:‘.’]	41 :	[ReferenceOP*ao:‘.’]
19 :	[Object*s:‘ObjApp’]	42 :	(Contain?ai?am)
20 :	[String*t:‘MAPI’]	43 :	-- skip-- (Argument?k?x)
21 :	[Method*u:‘GetNameSpace’]	44 :	-- skip-- (Argument?ab?aa)
22 :	[ReferenceOP*v:‘.’]	45 :	-- skip-- (Argument?an?a)
23 :	[Statement*w:‘Set’]	46 :	(Contain?ao?an)

Similarity of the source codes classified in each group was measured and compared with the "e-mail sending" malicious codes to evaluate maliciousness. For this purpose, each source code is expressed in a conceptual graph and changed into CGIF. Table 7 shows the results of similarity measured from the codes in Table 6 and changed source codes according to each group.

The results of similarity showed high discriminatory ability for the presence of malicious code. High similarity was detected in Group A and B. On the other hand, low similarity was detected in Group C containing normal behavior code not containing malicious codes, showing no malicious codes. Table 8 shows the results of comparing the proposed method with a vaccine program currently available.

Table 7. Similarity measurement in sample code according to each group

Groups compared	Similarity expressed in percentile
Group A	98%
Group B	83%
Group C	44%

Table 8. Results of comparing the proposed method with existing vaccines

	Proposed method		Vaccine from A company		Vaccine from B company	
	Detection	Warning	Detection	Warning	Detection	Warning
Group A	97	3	97	0	90	0
Group B	20	5	15	8	10	0
Group C	0	3	10	1	0	0

Although similar results were obtained for malicious codes falling into Groups A and C, the detection rate was high with the proposed method in this study for malicious codes in Group B. This result suggests that the method proposed in this paper is more feasible than the existing method of pattern matching.

6 Conclusion

In this study, malicious VBScript codes are expressed into conceptual graphs and their similarity was evaluated to detecting not only malicious codes but also modified malicious codes. According to the experiment done in this study, this method is effective in overcoming the problems with the existing methods, ie., the pattern matching or signature based detection, for detecting Internet worms having rapid spread speed and code change period. The method of detecting malicious codes by evaluating similarity based on conceptual graphs is effective for detecting not only modified malicious codes but also unknown malicious codes. Further studies are needed on the application of conceptual graphs by developing a tool for detection using the method of evaluating similarity and making definition on detailed concepts and relationship. The conceptual approach using conceptual graphs could be applied in the security area such as detecting intrusion. And also, it can be used in detecting malicious codes in the script-based programming environment of many kinds of embedded systems or telematics systems.

Acknowledgements

This research was supported by the MIC(Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Assessment). (IITA-2005-C1090-0502-0009)

References

1. Frithjof Dau. : Mathematical Foundations of Conceptual Graphs, 13th ICCS In Tutorial (2005)
2. O. Erdogan and P. Cao. Hash-av : Fast virus signature scanning by cache-resident filters, In http://crypto.stanford.edu/˜cao/hash-av/ (2005)

3. G. Mishne and M. de Rijke. : Source Code Retrieval using Conceptual Similarity, RIAO (2004) 539-554
4. Christodorescu, Jha. : Static Analysis of Executables to Detect Malicious Patterns, 12th USENIX Security Symposium, (2003)
5. Svetlana Hensman. : Construction of Conceptual Graph Representation of Texts, HLT-NAACL, (2004) 49-54
6. Karalopoulos, M. Kokla, M. Kavouras. : Geographic Knowledge Representation Using Conceptual Graphs, 7th AGILE Conference on Geographic Information Science, Crete, Greece, (2004)
7. J.-F. Baget. : Simple conceptual graphs revisited: Hypergraphs and conjunctive types for efficient projection algorithms, In Proc. of ICCS, (2003)
8. Jiwei Zhong, Haiping Zhu, Jianming Li and Yong Yu. : Conceptual Graph Matching for Semantic Search, In Proc. of ICCS, (2002)
9. Lei Zhang and Yong Yu. : Learning to Generate CGs from Domain Specific Sentences, In proc. of ICCS, LNAI 2120(Springer), (2001)
10. Harry S. Delugach : CharGer A Graphical Conceptual Graph Editor, In proc. of ICCS, (2001)
11. Pavlin Dobrev, Albena Strupchaska, Kristina Toutanova. : CGWorld-2001 - New Features and New Directions, In proc. of ICCS, (2001)
12. M. Montes y Gómez, A. Gelbukh, A. López López, Ricardo Baeza-Yates. : Flexible Comparison of Conceptual Graphs, In Proc. of DEXA-2001, (2001) 102-111
13. Francisco Fernandez. : Heuristic Engines, 11th International Virus Bulletin Conference, (2001)
14. Gabor Szappanos. : VBS Emulator Engine Design, Virus Bulletin Conference, (2001)
15. Igor Muttik. : Stripping down an AV Engines, Virus Bulletin Conference, (2000)
16. Montes y-Gómez, Gelbukh and López-López. : Comparison of Conceptual Graphs, Lecture Notes in Artificial Intelligence 1793, (2000)
17. 17. Sowa, John F. : Conceptual Graph Standard, American National Standard NCITS.T2/ISO/JTC1/SC32 WG2 N 0000. [Access Online : April 2001], URL : http://www.bestweb.net/~sowa/cg/cgstand.htm, (2001)
18. Sowa, John F. : Conceptual Structures Information Processing in Mind and Machine, Ed. Addison-Wesley, (1983)

A Minimized Test Pattern Generation Method for Ground Bounce Effect and Delay Fault Detection

MoonJoon Kim[1], JeongMin Lee[2], WonGi Hong[3], and Hoon Chang[4]

[1] Department of Computing, Graduate School, Soongsil University,
1-1, Sangdo-5Dong, Dongjak-Ku, Seoul, Korea
mjkim@watt.ssu.ac.kr
http://esrl.ssu.ac.kr

[2] Department of Computing, Graduate School, Soongsil University,
1-1, Sangdo-5Dong, Dongjak-Ku, Seoul, Korea
jmlee@watt.ssu.ac.kr
http://esrl.ssu.ac.kr

[3] Department of Computing, Graduate School, Soongsil University,
1-1, Sangdo-5Dong, Dongjak-Ku, Seoul, Korea
wghong@watt.ssu.ac.kr
http://esrl.ssu.ac.kr

[4] School of Computing, Soongsil University,
1-1, Sangdo-5Dong, Dongjak-Ku, Seoul, Korea
hoon@ssu.ac.kr

Abstract. An efficient board-level interconnect test algorithm is proposed considering both the ground bounce effect and the delay faults detection. The proposed algorithm is capable of IEEE 1149.1 interconnect test, negative ground bounce effect prevention, and also detects delay faults as well. The number of final test pattern set is not much different with the previous method, even our method enables to detect the delay faults in addition to the abilities the previous method guarantees.

1 Introduction

To test interconnect wire (nets) between two or more digital components, test engineers apply combinations of digital test stimuli to the net inputs, observe the responses at the net outputs, and compare them with expected responses. The literature offers many test generation algorithms for wiring interconnects. Their objective is to generate the smallest possible test pattern set, while guaranteeing certain detection and diagnostic properties. Fault models typically addressed by such algorithms are single-net opens and multiple-net shorts. The True/Complement test algorithm is able to detect the fault model and solve aliasing problem of test response. Also test pattern generation was proposed that detect delay faults on board level interconnection. The short/open on interconnection and delay faults detected using method that proposed in [5]. But proposed test pattern generation cannot detect incorrect operation of boundary scan follows in ground bounce.

Recently, preventing incorrect boundary scan test operation caused by ground bounce has become a new constraint for test generation algorithms. Ground bounce is the phenomenon of shifting ground and power voltage level. A method for preventing

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 577 – 583, 2006.

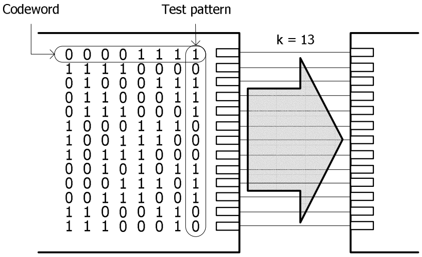
the negative effects of ground bounce is to place an upper limit, known as the simultaneously-switching-outputs limit (SSOL). Recently new test pattern generation algorithm for board level interconnection which considers this GB was developed. This technique is to test pattern selection minimizing count of simultaneously switching. And then if the number of simultaneously switching of assign pattern is more than SSOL, insert additional dummy pattern to generated test pattern set. It did so and modification of final test pattern it minimized. But this method has weak point that cannot detect delay faults.

This paper propose new test pattern set generate algorithm that can detect delay faults and support solution of incorrect operation problem by GB, contemporary interconnection test for board level test which guarantee previous technique of existing

2 Proposed Test Pattern Generation Method for Board Level Test

In this chapter, we propose test pattern generation method for board level test to detect delay faults contemporary consider GB effect. Previous method researched in [7] which base on True/Complement test algorithm. This algorithm divides 2steps. First step is codeword selection and second step is test pattern reordering.

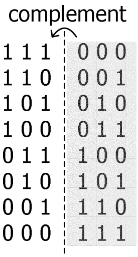
In first step, to test k interconnection we need $p(k) = 2\lceil \log_2 k \rceil$ test pattern.



```
0 0 0 0 1 1 1 1
1 1 1 1 0 0 0 0
0 1 0 0 0 1 1 1
0 1 1 0 0 0 1 1
0 1 1 1 0 0 0 1
1 0 0 0 1 1 1 0
1 0 0 1 1 1 0 0
1 0 1 1 1 0 0 0
0 0 1 0 1 0 1 1
0 0 0 1 1 1 0 1
0 0 1 1 1 0 0 1
1 1 0 0 0 1 1 0
1 1 1 0 0 0 1 0
```

Fig. 1. Test pattern set for interconnect test

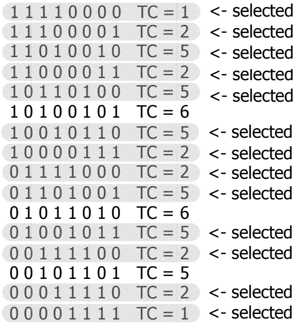
In Fig 1, the number of test pattern p(k) = 8 due to the number of interconnection is k = 13. This test pattern set can detect all of single-net opens and multiple-net shorts and solve the aliasing problem. In figure, each column called test pattern, each row called codeword.



```
1 1 1 | 0 0 0
1 1 0 | 0 0 1
1 0 1 | 0 1 0
1 0 0 | 0 1 1
0 1 1 | 1 0 0
0 1 0 | 1 0 1
0 0 1 | 1 1 0
0 0 0 | 1 1 1
```

Fig. 2. Characteristic of True/Complement algorithm

Existed method and propose method commonly base on True/Complement algorithm to generate candidate codewords. Basically the number of generated test pattern (p(k)) is 2⌈log$_2$k⌉, and even always. First, it reverses b generated test pattern and then creates another b test pattern. Whole test pattern generated by union this test pattern.



```
1 1 1 1 0 0 0 0  TC = 1  <- selected
1 1 1 0 0 0 0 1  TC = 2  <- selected
1 1 0 1 0 0 1 0  TC = 5  <- selected
1 1 0 0 0 0 1 1  TC = 2  <- selected
1 0 1 1 0 1 0 0  TC = 5  <- selected
1 0 1 0 0 1 0 1  TC = 6
1 0 0 1 0 1 1 0  TC = 5  <- selected
1 0 0 0 0 1 1 1  TC = 2  <- selected
0 1 1 1 1 0 0 0  TC = 2  <- selected
0 1 1 0 1 0 0 1  TC = 5  <- selected
0 1 0 1 1 0 1 0  TC = 6
0 1 0 0 1 0 1 1  TC = 5  <- selected
0 0 1 1 1 1 0 0  TC = 2  <- selected
0 0 1 0 1 1 0 1  TC = 5
0 0 0 1 1 1 1 0  TC = 2  <- selected
0 0 0 0 1 1 1 1  TC = 1  <- selected
```

Fig. 3. Candidate Codewords(k=13)

In figure 3 case, candidate codeword of 2^4=16 is created because the number (k) of interconnection is 13. In these things, the number of codeword finally to need is number *k* of them, number of interconnection. A switching-output about each codeword is viewed numerical value of TC in figure 3 and select a final codeword after selecting codeword *k* of them which is the most small. For this example, shaded codewords are selected codewords. If selecting for *k* of them interconnection is finished, go through a process of test pattern reordering.

Problem reordering test pattern by minimum cost is same as Salesman Problem. This NP-hard problem is brought solution by greedy heuristic algorithm. this is technique that choosing a minimum test pattern in added test patterns when select a next test pattern in selected test pattern after setting a starting test pattern in random. Do it repeatedly and then decide reordered sequence of assign about all test pattern using only minimum distance.

However in test pattern creating technique for existing board level interconnection, it has a weak point that cannot ensure delay faults test of interconnection so this paper propose that is to solve providing interconnection test in existing technique and also suggest a test pattern generation technique that can detect delay faults of interconnection perfectly in any case.

For testing about delay faults of interconnection, if each codeword to be applicable to test pattern occurs switching-outputs in 0 to 1. 1 to 0 at least one times, it is possible. In order to do this, this paper proposed to replace only process of the test pattern reordering, second step, algorithm considering existing GB problem. Pseudo-code of reordering process of technique by proposed in this paper is below.

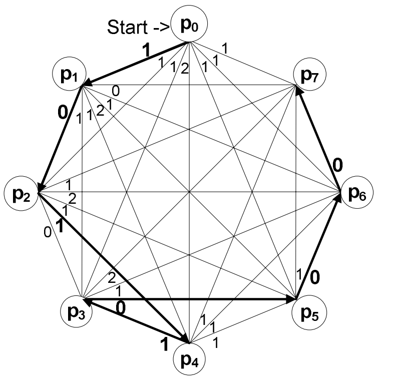
Proposed_Reorder_Algorithm

```
 1 :  $T$ = Completed test pattern set through codeword selection ;
 2 :  $D$ = Hamming distance of each test pattern ;
 3 :  $s$ = Value of SSOL ;
 4 :  $G$ = Draw_Graph ($T$, $D$, $s$) ;
 5 :  $T_R$ = Initialized reordered test pattern set ;
 6 :  Append ($T_R$, $p_0$) ;
 7 :  while ( !reordering complete )
 8 :  {
 9 :    if ( $b$-th test pattern)
10 :    {
11 :          while (Any_Continuous_Codeword_With_All_1_or_0 ($b$, $k$) == YES)
                  $x$ = selected test pattern by weighted ;
12 :               Append ($T_R$, $p_x$) ;
13 :    }
14 :    else
15 :    {
16 :          $x$ = selected test pattern that has the smallest weighted ;
17 :          Append ($T_R$, $p_x$) ;
18 :  }
19 :  return $T_R$ ;
```

First, draw undirected graph using test pattern set T, D which contains value of hamming distance value, and S means SSOL. (line 1-4) Each node is each test pattern. The edge that among the node, has the number of additional pattern $\lceil d/s \rceil$ - 1 to insert between test pattern. (line 4)

After select first test pattern to start node (line 6), all of test pattern reordering are completed after executes loop the number of remain test pattern p(k)-1 times. (line 7) While execute loop, all of test pattern select next test pattern using the smallest weighted of each node except b-th test pattern. (line 14-18) To reordered test pattern set detect delay faults, each codeword has from 0 to 1, from 1 to 0 at least one. While from first pattern to b-1 th test pattern, we check each codeword has continuous 0's or 1's and then repeat compare assignability of test pattern that has priority from high to low in second while statement. (line 11-12) We insert b-th test pattern to test pattern set, if selected test pattern can detect delay faults. Since b+1 th test pattern, we assign test pattern which has the smallest value basically. Eventually terminate propose algorithm, due to return completed reordered test pattern $T_{R.}$

Figure 4 (a) show reordering process, that consider GB effect and delay faults from first step completed test pattern. Shown as figure, test pattern that through propose algorithm has from 0 to 1, 1 to 0 switching-output at least one. Consequently test pattern set execute delay faults test perfectly. Test Pattern generation of propose algorithm base on greedy heuristic algorithm [8], the maximum, minimum value that test pattern has, is same as previous algorithm. [7]



(a)

	p_7	p_6	p_5	p_3	p_4	p_2	p_1	p_0
c_0	0	0	0	1	0	1	1	1
c_1	1	1	1	0	1	0	0	0
c_2	1	0	0	0	0	1	1	1
c_3	1	1	0	0	0	0	1	1
c_4	1	1	1	0	0	0	0	1
c_5	0	0	0	1	1	1	1	0
c_6	0	0	1	1	1	1	0	0
c_7	0	1	1	1	1	0	0	0
c_8	0	1	0	1	0	0	1	1
c_9	0	0	1	1	0	1	0	1
c_{10}	0	1	1	1	0	0	0	1
c_{11}	1	0	0	0	1	1	1	0
c_{12}	1	1	0	0	1	0	1	0

5 5 5 7 6 5 6

Hamming Distance

(b)

Fig. 4. Proposed test pattern reordering method for delay faults model (k=13, s=5)

Figure 4 (b) show hamming distance that calculated from reordered test pattern set. There are 3 shade test patterns shown as figure 4 (b). The one more test pattern that insert to previous test pattern set, is use to detect delay faults.

	p_7	p_6	p_5	p_3	$p_{/2}$	p_4	$p_{/1}$	p_2	p_1	$p_{/0}$	p_0
c_0	0	0	0	1	1	0	0	1	1	1	1
c_1	1	1	1	0	0	1	0	0	0	0	0
c_2	1	0	0	0	0	0	1	1	1	1	1
c_3	1	1	0	0	0	0	0	0	1	1	1
c_4	1	1	1	0	0	0	0	0	0	0	1
c_5	0	0	0	1	1	1	1	1	1	0	0
c_6	0	0	1	1	1	1	1	1	0	0	0
c_7	0	1	1	1	1	1	0	0	0	0	0
c_8	0	1	0	1	0	0	0	0	1	1	1
c_9	0	0	1	1	0	0	1	1	0	1	1
c_{10}	0	1	1	1	0	0	0	0	0	1	1
c_{11}	1	0	0	0	1	1	1	1	1	0	0
c_{12}	1	1	0	0	1	1	0	0	1	0	0

5 5 5 5 2 5 1 5 5 1

Hamming Distance

Fig. 5. dummy test pattern insertion to the generated test pattern set from proposed method

Fig 5 is process of insert additional test pattern to generated test pattern set caused by proposed method. Test pattern set shown as Fig 4 detect interconnection test and test delay faults further more reordered due to reduce negative ground bounce effect. But there is test pattern that has switching-outputs more than SSOL. We can solve incorrect boundary scan operation by insert 3 dummy test patterns like shaded part of figure 5.

Previous board level test is only considers interconnection test and negative ground bounce effect. Eventually Compare to previous method, proposed generate test pattern method is more efficient which can interconnection test and prevent incorrect boundary scan operation by reduce negative ground bounce effect additional detect delay faults.

3 Comparison with Previous Work and Experiment Result

We experimented and compared result on various number of SSOL when the number of interconnection is 8000(a), 7000(b), 6000(c) and 5000(d). S in first column that means value of SSOL, second column is the number of test pattern in previous method which only consider negative GB effect and third column is proposed method that is capable of IEEE 1149.1 interconnect test, negative ground bounce effect prevention, and also detects delay faults as well. There are total 26 test patterns was required in every case of experiment environment (k=8000, 7000, 6000, 5000) at method which considered negative ground bounce effect except delay faults.

Table 1. Total number of test pattern Comparison

k = 8000

SSOL (*s*)	Number of test pattern	
	EXTEST + GB	EXTEST + GB + Delay
400	251	251
800	126	126
1200	101	101
1600	76	76
2000	51	51
2400	51	51
2800	51	51
3200	51	51
3600	51	51
4000	26	26

(a)

k = 7000

SSOL (*s*)	Number of test pattern	
	EXTEST + GB	EXTEST + GB + Delay
350	251	252
700	126	127
1050	101	101
1400	76	76
1750	51	52
2100	51	51
2450	51	51
2800	51	51
3150	51	51
3500	26	27

(b)

k = 6000

SSOL (*s*)	Number of test pattern	
	EXTEST + GB	EXTEST + GB + Delay
300	227	234
600	126	127
900	77	81
1200	76	76
1500	51	52
1800	51	51
2100	51	51
2400	51	51
2700	27	29
3000	26	27

(c)

k = 5000

SSOL (*s*)	Number of test pattern	
	EXTEST + GB	EXTEST + GB + Delay
250	227	238
500	127	**126**
750	77	81
1000	76	76
1250	52	**51**
1500	51	51
1750	51	51
2000	51	51
2250	27	29
2500	27	**26**

(d)

This proposed method finds previous interconnection test and negative ground bounce effect and additional detects delay faults. Table 1 shows comparative number of test pattern. In (a), (b) and (c), we can see that the proposed method is need equal or not much different the number of test pattern of previous method. Moreover, (d) shows that number of test pattern of proposed method is sometimes smaller than previous method.

4 Conclusions

In this paper, we proposed efficient test algorithm of board-level that can prevent incorrect boundary scan test operation caused by ground bounce and detect interconnect delay test perfectly. The number of total test patterns generated by the proposed algorithm is shown to be similar to previous method, even our method enables to detect the delay faults in addition to the abilities the previous method guarantees.

Acknowledgement

This work was supported by the Soongsil University Research Fund

Reference

1. IEEE Std. 1149.1-2001, Test Access Port and Boundary Scan Architecture, IEEE, 2001.
2. W. H. Kautz, "Testing of Faults in Wiring Interconnects," IEEE Trans. Computers, vol. 23, no. 4, 1974.
3. P. Goel and M. T. Mcmahon, "Electronic Chip-in-Place Test," Proc. Int'l Test Conf., 1982.
4. J. T. de Sousa and P. Y. K. Cheung, Boundary Scan Interconnect Diagnosis, Kluwer Academic, 2001.
5. S Park, T Kim, "A New IEEE 1149.1 Boundary Scan Design for the Detection of Delay Defects", Design, Automation and Test in Europe Conf., 2000.
6. H. D. L. Hollmann, E. J. Marinissen, B. Vermeulen, "Optimal interconnect ATPG under a ground-bounce constraint,", Proc. Int'l Test Conf., pp. 60-69, 2003.
7. E. J. Marinissen, R. G. Bennetts, "Minimizing Pattern Count for Interconnect Test under a Ground Bounce Constraint," IEEE Design & Test of Computers, Vol. 20, Issue 2, pp. 8-19, 2003.
8. D. S. Johnson and L. A. McGeoch, "The Traveling Salesman Problem: A Case Study," Local Search in Combinatorial Optimization, E. H. L. Aarts and J.-K. Lensta, eds., John Wiley & Sons, 1997.

Efficient Exponentiation in $GF(p^m)$ Using the Frobenius Map*

Mun-Kyu Lee[1,**], Howon Kim[2], Dowon Hong[2], and Kyoil Chung[2]

[1] School of Computer Science and Engineering,
Inha University, Incheon 402-751, Korea
mklee@inha.ac.kr
[2] Electronics and Telecommunications Research Institute,
161 Gajeong-dong, Yuseong-gu,
Daejeon, 305-350, Korea

Abstract. The problem of exponentiation over a finite field is to compute A^e for a field element A and a positive integer e. This problem has many useful applications in cryptography and information security. In this paper, we present an efficient exponentiation algorithm in optimal extension field (OEF) $GF(p^m)$, which uses the fact that the Frobenius map, i.e., the p-th powering operation is very efficient in OEFs. Our analysis shows that the new algorithm is twice as fast as the conventional square-and-multiply exponentiation. One of the important applications of our new algorithm is random generation of a base point for elliptic curve cryptography, which is an attractive public-key mechanism for resource-constrained devices. We present a further optimized exponentiation algorithm for this application. Our experimental results show that the new technique accelerates the generation process by factors of 1.62–6.55 over various practical elliptic curves.

Keywords: Cryptography, Exponentiation, Finite Field, Optimal Extension Field, Elliptic Curve.

1 Introduction

The problem of exponentiation in the Galois field $GF(p^m)$ is to compute $A^e \in GF(p^m)$ when $A \in GF(p^m)$ and a positive integer e are given, where p is a prime and $m \geq 1$. This problem has many important applications in cryptography and information security, and it is crucial to develop an efficient exponentiation algorithm. Many researchers have proposed a number of methods to speed up exponentiation [1]. Especially, there are various methods using a normal basis representation, where a p-th powering operation is almost free [2, 3, 4].

In this paper, we show that similar approach is possible also in a polynomial basis representation, if the underlying field $GF(p^m)$ is an optimal extension field (OEF) [5, 6]. Using the fact that p-th powering is a very fast operation in OEFs,

* This work was supported by INHA UNIVERSITY Research Grant.

** Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 584–593, 2006.

we first present an efficient exponentiation algorithm over OEF. Our analysis and experimental results show that the new algorithm is approximately twice as fast as the well-known square-and-multiply algorithm.

One of the important applications of our new algorithm is the square root computation over OEF and random generation of a base point over an elliptic curve defined over OEF. We present a further optimized exponentiation algorithm for this application. Our experimental results show that the new algorithm accelerates the point generation process by factors of 1.62–6.55 over various practical elliptic curves.

2 Optimal Extension Field

2.1 Optimal Extension Field (OEF)

An OEF [5, 6] is a finite field $GF(p^m)$ that satisfies the following:

1. p is a prime less than but close to the word size of the target processor.
2. $p = 2^n - c$, where $\log_2 c \leq n/2$, and
3. An irreducible binomial $f(x) = x^m - w$ exists.

For a cryptographic application, an odd prime m is used [7]. An element A in an OEF is represented as $A = \sum_{i=0}^{m-1} a_i x^i$ using the polynomial basis representation, where $a_i \in GF(p)$. An OEF enables us to exploit the full computing power of a general purpose processor in software implementation of elliptic curve cryptosystems (ECCs).

2.2 Multiplication and Squaring in OEF

A multiplication in OEF is composed of two stages [5, 6]. The first one is an ordinary polynomial multiplication of two OEF elements A and B, producing an intermediate product C' of degree less than or equal to $2m-2$. The schoolbook method to calculate the coefficients of C' requires m^2 multiplications and $(m-1)^2$ additions in the subfield $GF(p)$.

The second stage is the reduction stage where $C' \bmod f(x)$ is calculated to get $C = A \cdot B \bmod f(x) \in GF(p^m)$. By $x^m \equiv w \bmod f(x)$, one can do this using $m-1$ multiplications and $m-1$ additions in $GF(p)$.

Therefore, one multiplication in $GF(p^m)$ requires $m^2 + m - 1$ multiplications and $m^2 - m$ additions in $GF(p)$.

In the case of squaring, the only difference is that the number of coefficient multiplications in the first stage is reduced to $m(m+1)/2$. Hence, the required number of $GF(p)$ operations is $m^2/2 + 3m/2 - 1$ multiplications and $m^2 - m$ additions in total.

Note that a multiplication in $GF(p)$ is much more expensive than an addition in $GF(p)$ on most of the current general purpose CPUs. Hence we will ignore the cost for additions in $GF(p)$ throughout this paper.

Algorithm 1. Square-and-multiply exponentiation

```
1. $B \leftarrow 1$.
2. for $i$ from $l-1$ to 0 do
3.     $B \leftarrow B^2$.
4.     if $(e_i = 1)$ then $B \leftarrow B \cdot A$.
5. od
6. output $B$.
```

2.3 Frobenius Map Computation in OEF [8]

For $A = \sum_{i=0}^{m-1} a_i x^i \in GF(p^m)$, the Frobenius map, i.e., the p-th power is represented as $a_0 + \sum_{i=1}^{m-1} a_i x^{ip}$ since $a_i^p = a_i$ in $GF(p)$. By $x^m \equiv w \bmod f(x)$, we get

$$x^{ip} \equiv x^{(ip \bmod m)} w^{\lfloor ip/m \rfloor} \bmod f(x)$$

for $1 \le i \le m-1$, and

$$A^p = a_0 + \sum_{i=1}^{m-1} a_i w^{\lfloor ip/m \rfloor} x^{(ip \bmod m)}.$$

After reordering terms, we obtain a polynomial basis representation of A^p. Note that we can pre-compute $w^{\lfloor ip/m \rfloor}$, since p, m, and w are independent of A. Hence, a p-th powering operation can be done by only $m-1$ on-line multiplications in $GF(p)$.

2.4 Square-and-Multiply Exponentiation in OEF

There are numerous exponentiation algorithms suggested so far. Among them, the most simple and powerful algorithm is the square-and-multiply algorithm [1]. Although other algorithms such as the m-ary method, the sliding window method and the addition chain method show better performance, their performance gain over the square-and-multiply algorithm is not greater than 20–30%. Therefore, in this paper, we will take the square-and-multiply algorithm as a standard to measure the performance of our new exponentiation algorithms.[1]

For an element $A \in GF(p^m)$ and an integer $e = \sum_{i=0}^{l-1} e_i 2^i$, where $l = \lceil \log_2(e+1) \rceil$ and $e_i \in \{0, 1\}$, Algorithm 1 computes A^e using the left-to-right square-and-multiply method.

If we assume that line 4 is executed $l/2$ times on the average, the number of multiplications in $GF(p)$ in Algorithm 1 is

$$l(m^2/2 + 3m/2 - 1) + l(m^2 + m - 1)/2 = l(m^2 + 2m - 3/2), \tag{1}$$

according to the estimation given in Section 2.2.

[1] If we use the property of a specific problem instance of exponentiation, we can obtain more speedup. For example, if element A is fixed and only e varies in the repeated computation of A^e, then we can pre-compute some information on A and significantly reduce the amount of on-line computation. This technique, however, cannot be applied to a situation where A is algo a variable.

Algorithm 2. Finding a random point G of order r

```
1. Generate a random point $P$ (not $O$) on $E$ as follows:
   1.1 Choose random $X \in GF(p^m)$.
   1.2 Set $Z \leftarrow X^3 + AX + B$.
   1.3 If $Z = 0$, then go to step 1.1.
   1.4 If a square root of $Z$ exists, then set it as $Y$; otherwise, go to step 1.1.
   1.5 Set $P \leftarrow (X, Y)$.
2. Set $G \leftarrow hP$.
3. If $G = O$, then go to step 1.
4. Output $G$.
```

3 Exponentiation and Its Application to Elliptic Curves

In this section, we examine the relation between OEF exponentiation and elliptic curve point generation. An elliptic curve over $GF(p^m)$ is given by

$$E : Y^2 = X^3 + AX + B,$$

where $A, B \in GF(p^m)$ and $4A^3 + 27B^2 \neq 0$. It is well known that E forms an additive group under point addition operation.

If we select A and B from $GF(p)$ and we still consider X and Y over $GF(p^m)$, then elliptic curve operations, especially scalar multiplications, can be accelerated. We will call this kind of curve a Koblitz-type curve which is a generalization of Koblitz curve with $p = 2$. For distinction, we will call a curve with $A, B \in GF(p^m)$ a random curve.

For cryptographic applications, an elliptic curve is chosen so that its group order may be divisible by a sufficiently large prime, i.e., order $= hr$ for a large prime r and a small integer h [9]. (We call h the cofactor.) Then all cryptographic protocols are performed over a subgroup generated by a point G of order r, which is called a base point. Hence, it is crucial to find an adequate base point in cryptographic applications.

Algorithm 2 shows an algorithm that finds a random base point G [7, 9]. In this algorithm, the most time-consuming part is the computation of a square root in line 1.4. (Note that this part may be performed several times.) The cost to compute hP (line 2) is negligible for a typical random curve, since a curve is selected so that $h = 1$ in this case. For a Koblitz-type curve, h is selected so that $h \approx p$. Hence the cost for line 2 is comparable to that of line 1.4 in this case.

To compute a square root, we use Algorithm 3 or 4 according to the form of p^m [7, 9]. (We omit the case $p^m \equiv 1 \bmod 8$ since our exponentiation algorithm does not apply to this case.) Note that exponentiation over $GF(p^m)$ (line 2 of Algorithms 3 and 4) is the only significant operation in the square root computation from the viewpoint of execution time.[2]

By the above discussion, we see that an efficient exponentiation over $GF(p^m)$ is very important for efficient generation of a base point.

[2] Algorithms 3 and 4 are from elliptic curve standards [7] and [9]. Note that there are also other algorithms to compute square root [10, 11].

Algorithm 3. Computing a square root of $Z \in GF(p^m)$ $(p^m \equiv 3 \bmod 4)$

```
1. Set $u \leftarrow (p^m - 3)/4$.
2. Compute $Y \leftarrow Z^{u+1} \in GF(p^m)$.
3. If $Y^2 = Z$, then output $Y$; otherwise no square root exists.
```

Algorithm 4. Computing a square root of $Z \in GF(p^m)$ $(p^m \equiv 5 \bmod 8)$

```
1. Set $u \leftarrow (p^m - 5)/8$.
2. Compute $B \leftarrow (2Z)^u \in GF(p^m)$.
3. Compute $C \leftarrow 2ZB^2$.
4. Compute $Y \leftarrow ZB(C - 1)$.
5. If $Y^2 = Z$, then output $Y$; otherwise no square root exists.
```

4 New Exponentiation Algorithm in OEF

In this section, we give a new algorithm to compute $A^e \in GF(p^m)$. In our algorithm,

- the exponent e is regarded as a p-ary number, i.e., $e = \sum_{i=0}^{s-1} e'_i p^i$, where $s = \lceil \log_p(e+1) \rceil$ and $0 \le e'_i \le p-1$, and
- each coefficient e'_i is regarded as a binary number, i.e., $e'_i = \sum_{j=0}^{t-1} e'_{ij} 2^j$, where $t = \lceil \log_2(p+1) \rceil$ and $e'_{ij} \in \{0, 1\}$.

Thus we can see A^e as

$$\begin{aligned} A^e &= \left(A^{p^{s-1}}\right)^{e'_{s-1}} \times \left(A^{p^{s-2}}\right)^{e'_{s-2}} \times \cdots \times A^{e'_0} \\ &= \left(A^{p^{s-1}}\right)^{e'_{s-1,t-1}2^{t-1} + e'_{s-1,t-2}2^{t-2} + \cdots + e'_{s-1,0}} \\ &\quad \times \left(A^{p^{s-2}}\right)^{e'_{s-2,t-1}2^{t-1} + e'_{s-2,t-2}2^{t-2} + \cdots + e'_{s-2,0}} \\ &\quad \vdots \\ &\quad \times A^{e'_{0,t-1}2^{t-1} + e'_{0,t-2}2^{t-2} + \cdots + e'_{0,0}}. \end{aligned}$$

Now we can compute A^e in two stages. In the first stage, we construct a p^i-th power table T, i.e., $T_i = A^{p^i}$ for $0 \le i \le s-1$. Here the p-th powering operations are used $s-1$ times. The second stage is a simultaneous square-and-multiply exponentiation of $(T_{s-1})^{e'_{s-1}}, (T_{s-2})^{e'_{s-2}}, \ldots, (T_0)^{e'_0}$. Algorithm 5 shows the complete procedure.

The most time-consuming parts of Algorithm 5 are lines 1, 4 and 6. (The execution time for other parts can be ignored.) We estimate the amount of required computation for these lines as follows:

- Since a p-th powering operation is done by $m-1$ multiplications in $GF(p)$, the total amount of computation in line 1 is

$$(s-1)(m-1) \tag{2}$$

Algorithm 5. Exponentiation in $GF(p^m)$ using efficient p-th powering

```
1. $T_i \leftarrow A^{p^i}$ for $0 \le i \le s-1$.
2. $B \leftarrow 1$.
3. for $j$ from $t-1$ to 0 do
4.    $B \leftarrow B^2$.
5.    for $i$ from $s-1$ to 0 do
6.       if $(e'_{ij} = 1)$ then $B \leftarrow B \cdot T_i$.
7.    od
8. od
9. output $B$.
```

multiplications in $GF(p)$.

- Line 4 is executed t times. Hence, according to the estimation given in Section 2.2, the required number of multiplications in $GF(p)$ is
$$t(m^2/2 + 3m/2 - 1). \tag{3}$$
- Assuming the half of e'_{ij}'s are one as in the square-and-multiply algorithm, we see that line 6 is executed $st/2$ times. Hence, the required number of multiplications in $GF(p)$ is
$$st(m^2 + m - 1)/2. \tag{4}$$

5 Optimized Exponentiation for Elliptic Curve Point Generation

In this section we show that our new exponentiation algorithm can be optimized further if it is used to compute a square root and to generate a base point over an elliptic curve. We will use the fact that the exponents u in Algorithms 3 and 4 are fixed and they have special structures.

First, we consider the case that $p^m \equiv 3 \bmod 4$ (Algorithm 3). Note that $p^m - 3$ can be represented as an m-digit p-ary number, i.e.,

$$(p-1, p-1, \ldots, p-1, p-3),$$

in a vector representation. This can be rewritten as

$$(p-3, 3p-1, \ldots, p-3, 3p-1, p-3), \tag{5}$$

since $(p-1)p + (p-1) = (p-3)p + (3p-1)$ and m is odd. Because $p^m \equiv 3 \bmod 4$ implies $p \equiv 3 \bmod 4$ and $3p \equiv 1 \bmod 4$, (5) is divisible by 4. Therefore we obtain

$$u = (p^m - 3)/4 = ((p-3)/4, (3p-1)/4, \ldots, (p-3)/4, (3p-1)/4, (p-3)/4)\,.$$

Hence we can compute A^u as

$$A^u = \left(A \times A^{p^2} \times \cdots \times A^{p^{m-1}}\right)^{(p-3)/4} \times \left(A^p \times A^{p^3} \times \cdots \times A^{p^{m-2}}\right)^{(3p-1)/4}.$$

Algorithm 6. Optimized computation of A^u ($p^m \equiv 3 \bmod 4$ or $p^m \equiv 5 \bmod 8$)

```
1. if ($p \equiv 3 \bmod 4$) then $e_0 \leftarrow (p-3)/4$; $e_1 \leftarrow (3p-1)/4$.
   else $e_0 \leftarrow (p-5)/8$; $e_1 \leftarrow (5p-1)/8$.
2. $T_0 \leftarrow A \times A^{p^2} \times \cdots \times A^{p^{m-1}}$; $T_1 \leftarrow A^p \times A^{p^3} \times \cdots \times A^{p^{m-2}}$.
3. $B \leftarrow 1$.
4. for $j$ from $t-1$ to 0 do
5.     $B \leftarrow B^2$.
6.     if ($e_{0j} = 1$) then $B \leftarrow B \cdot T_0$.
7.     if ($e_{1j} = 1$) then $B \leftarrow B \cdot T_1$.
8. od
9. output $B$.
```

Similarly, for the case that $p^m \equiv 5 \bmod 8$ (Algorithm 4), we can compute A^u as

$$A^u = \left(A \times A^{p^2} \times \cdots \times A^{p^{m-1}}\right)^{(p-5)/8} \times \left(A^p \times A^{p^3} \times \cdots \times A^{p^{m-2}}\right)^{(5p-1)/8}.$$

Algorithm 6 shows the complete procedure for these two cases, where $t = \lceil \log_2(p+1) \rceil$.

Now we count the number of required operations.

- To compute T_1 in line 2, we need $(m-2)$ p-th powering operations and $(m-3)/2$ multiplications over $GF(p^m)$. T_0 can be computed using one p-th powering operation and one multiplication, since $T_0 = A \times T_1^p$.
- Line 5 is executed t times. Hence we need t squarings.
- Lines 6 and 7 are executed t times in the worst case. We can't use the estimation that the half of e_{0j}'s and e_{1j}'s are ones on the average, since e_0 and e_1 are fixed for a specific p and they have much more ones than the average value. (This is because p has been selected so that it may contain many ones in the binary representation to satisfy the OEF property 2 given in Section 2.1.)

By the estimation given in Section 2.2 and Section 2.3, the worst-case analysis shows that the total number of multiplications in $GF(p)$ is

$$\begin{aligned}&(m-1)\left((m-2)+1\right)+(m^2+m-1)((m-3)/2+1+2t)+(m^2/2+3m/2-1)t \\ &= (m^3+2m^2-6m+3)/2 + t(5m^2+7m-6)/2 \qquad (6)\end{aligned}$$

6 Comparison of Efficiency

In this section, we compare the computational costs of the square-and-multiply algorithm and the two new exponentiation algorithms. For many cryptographic applications including base point generation, the size of exponent in exponentiation is approximately the same as the group order, i.e., $e \approx p^m$ in our context. Then l in (1) and s in (2) and (4) satisfies

$$l = \lceil \log_2(e+1) \rceil \approx m \log_2 p \quad \text{and} \quad s = \lceil \log_p(e+1) \rceil \approx m,$$

Table 1. Number of $GF(p)$ multiplications to compute $A^e \in GF(p^m)$

algorithms	number of multiplications in $GF(p)$
square-and-multiply (Alg. 1)*	$\frac{\log_2 p}{2}(2m^3+4m^2-3m)$
new algorithm (Alg. 5)*	$(m-1)^2+\frac{\log_2 p}{2}(m^3+2m^2+2m-2)$
optimized algorithm for $e=u$ (Alg. 6)†	$\frac{m^3+2m^2-6m+3}{2}+\frac{\log_2 p}{2}(5m^2+7m-6)$

*average-case analysis, †worst-case analysis

respectively. Also we can use an approximation $t \approx \log_2 p$ in (3), (4) and (6). Then we can estimate the number of multiplications in $GF(p)$ for exponentiation of an element in $GF(p^m)$. See Table 1.

From Table 1, we can see that our general exponentiation algorithm given in Section 4 requires about a half of computation compared to the square-and-multiply algorithm, and the optimized algorithm given in Section 5 further reduces the amount of computation by eliminating the factor $\log_2 p$ in the leading term.

The number of $GF(p)$ multiplications of Algorithm 6 can be written as $O(m^2 (m+\log_2 p))$ using an asymptotic notation. We remark that this bound can be improved to $O(m^2(\log_2 m+\log_2 p))$ if we use the addition chain technique [10, 11] to compute T_0 and T_1 in line 2 of Algorithm 6. Note that the bound $O(m^2(\log_2 m+\log_2 p))$ is the same as those of customized algorithms for square root computation given in [10, 11], which are not derived from general exponentiation, but designed for the specific purpose of square root computation.

We also remark that Algorithms 5 and 6 require only small amount of additional memory to store T_i's. (Although we should also store $w^{\lfloor ip/m \rfloor}$ values for p-th powering, it is not an overhead; the code for p-th powering is necessary for other operations such as a field inversion and elliptic curve point operations, regardless of the use of new exponentiation algorithm.)

7 Experimental Results

To verify the estimation given in the previous section, we implemented various OEFs and elliptic curves, and we measured the timings for exponentiation and point generation. First, Table 2 shows the OEFs and curves which we have implemented.

Table 3 shows the measured timings for an exponentiation A^e with $e \approx p^m$. We implemented the algorithms in C using djgpp-2.03 compiler on a Pentium 4 2.66GHz CPU. In Table 3, we first see that Algorithm 5 is about twice as fast as the square-and-multiply algorithm (Algorithm 1), which is consistent with the estimation given in the previous section. We also see that computational speedups obtained using Algorithm 6 are greater than the estimation given in Table 1. Note that we have used a worst-case analysis for Algorithm 6 in Table 1.

Next, we measured the timings to produce a random base point using Algorithm 2. See Table 4. In this table, we see that if we use Algorithm 5 instead of

Table 2. Implemented OEFs and curves

OEF: $GF(p^m)$	curve: $Y^2 = X^3 + AX + B$
OEF 1[†]: $p = 2^{16} - 129$ $m = 11$ $f(x) = x^{11} - 3$	curve 1R: random curve with 176-bit order, cofactor $h = 1$ [7] $A =$ 0X FF7C, $B =$ 0X 325Ax^{10}+5511x^9+F0A7x^8+B7FBx^7+D906x^6+1FBAx^5 +D032x^4+CC2Dx^3+EE25x^2+C40Ax+ECAF
	curve 1K: Koblitz-type curve with 160-bit order, $h = 65407$ [7] $A =$ 0X FF7C, $B =$ 0X 017F
OEF 2[†]: $p = 2^{16} - 17$ $m = 17$ $f(x) = x^{17} - 2$	curve 2R: random curve with 272-bit order, $h = 1$ [7] $A =$ 0X FFEC, $B =$ 0X C3EDx^{16}+AB1Fx^{15}+5ED9x^{14}+2A01x^{13}+ACDEx^{12} +3D1Ex^{11}+A38Dx^{10}+5A95x^9+9D10x^8+1F9Ex^7+5C63x^6 +86B7x^5+7F7Ax^4+66C1x^3+6159x^2+947Fx+4B36
OEF 3[‡]: $p = 2^{15} - 75$ $m = 11$ $f(x) = x^{11} - 2$	curve 3K: Koblitz-type curve with 150-bit order, $h = 32420$ $A =$ 0X 0001, $B =$ 0X 0000
OEF 4[†]: $p = 2^{31} - 1$ $m = 7$ $f(x) = x^7 - 3$	curve 4R: random curve with 217-bit order, $h = 1$ [7] $A =$ 0X 7FFFFFFC, $B =$ 0X 039055B8x^6+1A52D0E2x^5+2EEE1471x^4+07505B48x^3 +6A6BFE64x^2+4C1292C9x+36BB468C
	curve 4K: Koblitz-type curve with 187-bit order, $h = 2147444533$ $A =$ 0X 00000000, $B =$ 0X 00000005
OEF 5[‡]: $p = 2^{29} - 3$ $m = 7$ $f(x) = x^7 - 2$	curve 5K: Koblitz-type curve with 162-bit order, $h = 3563249795090$ $A =$ 0X 00000002, $B =$ 0X 00000000

[†] $p^m \equiv 3 \bmod 4$, [‡] $p^m \equiv 5 \bmod 8$

Table 3. Timings for the computation of A^e with $e \approx p^m$ (μsec)

OEF	Algorithm 1	Algorithm 5 (random e)		Algorithm 6 (e fixed as u)	
	timings (A)	timings (B)	speedups (A)/(B)	timings (C)	speedups (A)/(C)
1	286	130	2.20	57	5.02
2	884	408	2.17	125	7.07
3	390	199	1.96	70	5.57
4	186	97	1.92	58	3.21
5	470	199	2.36	124	3.79

Algorithm 1, then we obtain minor speedups. (The speedups are much smaller than those expected from Table 3, i.e., approximately two for random curves. Note that the real exponents u have special structures which are explained in Section 5 and they have larger Hamming weights than average.) The speedups obtained by using Algorithm 6, however, are significant: we can produce a random base point by 1.62 to 6.55 times faster. Note that the speedups on random

Table 4. Timings to produce a random base point (μsec)

curve	using Alg. 1	using Alg. 5		using Alg. 6	
	timings (A)	timings (B)	speedups (A)/(B)	timings (C)	speedups (A)/(C)
1R	570.1	417.2	1.37	129.6	4.40
1K	848.2	635.9	1.33	359.0	2.36
2R	1908.0	1294.2	1.47	291.5	6.55
3K	1168.5	813.4	1.44	525.9	2.22
4R	324.7	255.1	1.27	113.3	2.87
4K	580.3	516.5	1.12	396.7	1.46
5K	1521.4	1397.6	1.09	937.6	1.62

curves are much greater than those on Koblitz-type curves, since the cost to compute hP in Algorithm 2 is not negligible in Koblitz-type curves, and it is an irreducible overhead.

References

1. Gordon, D.M.: A survey of fast exponentiation methods. Journal of Algorithms **27** (1998) 129–146
2. Agnew, G.B., Mullin, R.C., Vanstone, S.A.: Fast exponentiation in GF(2^n). In: EUROCRYPT '88. Volume 330 of LNCS., Springer (1988) 251–256
3. von zur Gathen, J.: Processor-efficient exponentiation in finite fields. Information Processing Letters **41** (1992) 81–86
4. Lee, M.K., Kim, Y., Park, K., Cho, Y.: Efficient parallel exponentiation in $GF(q^n)$ using normal basis representations. Journal of Algorithms **54** (2005) 205–221
5. Bailey, D.V., Paar, C.: Optimal extension fields for fast arithmetic in public-key algorithms. In: CRYPTO '98. Volume 1462 of LNCS., Springer (1998) 472–485
6. Bailey, D.V., Paar, C.: Efficient arithmetic in finite field extensions with application in elliptic curve cryptography. Journal of Cryptology **14** (2001) 153–176
7. TTAS.KO-12.0015: Digital Signature Mechanism with Appendix– Part 3: Korean Certificate-based Digital Signature Algorithm using Elliptic Curves. (2001)
8. Kobayashi, T.: Base-ϕ method for elliptic curves of OEF. IEICE Trans. Fundamentals **E83-A** (2000) 679–686
9. IEEE P1363-2000: IEEE Standard Specifications for Public-Key Cryptography. (2000)
10. Barreto, P.S., Kim, H.Y., Lynn, B., Scott, M.: Efficient algorithms for pairing-based cryptosystems. In: CRYPTO 2002. Volume 2442 of LNCS., Springer (2002) 354–369
11. Feng, W., Nogami, Y., Morikawa, Y.: A fast square root computation using the Frobenius mapping. In: ICICS 2003. Volume 2836 of LNCS., Springer (2003) 1–10

A Dual-Channel MAC Protocol Using Directional Antennas in Location Aware Ad Hoc Networks[*]

DoHyung Han[1], JeongWoo Jwa[1], and HanIl Kim[2]

[1] Department of Telecommunication Engineering, Cheju National University,
1 Ara-dong Jeju, Korea
{figure21, lcr02}@cheju.ac.kr
[2] Department of Computer Education, Cheju National University,
1 Ara-dong Jeju, Korea
hikim@cheju.ac.kr

Abstract. Ad hoc MAC protocols using directional antennas can be used to improve the network capacity by improving spatial reuse. But, directional MAC protocols have the problem of deafness and have a poor throughput performance. The dual-channel MAC protocol with an omnidirectional antenna has been proposed to mitigate deafness. In this paper, we propose a dual-channel MAC protocol using the omnidirectional antenna for control channel and directional antennas for data channel. In the proposed MAC protocol, the omnidirectional antenna used in control channel mitigates deafness and directional antennas used in data channel improve spatial reuse. The throughput performance of the proposed MAC protocol is confirmed by computer simulations using Qualnet ver. 3.8 simulator.

1 Introduction

Ad hoc MAC protocols using directional antennas have been proposed to improve the network capacity [1-7]. Directional MAC protocols have advantages of improving spatial reuse and reducing power consumption. But, directional RTS (DRTS) causes deafness because node in directional transmission does not hear and respond to RTS from other nodes in the coverage. The transmitting node retransmits RTS to node in deafness and the backoff period of the node is exponentially increased at every retransmission. Therefore, deafness caused by DRTS degrades the throughput performance of ad hoc networks. In the DMAC protocol [2], the blocking algorithm for directional antennas should be used to improve spatial reuse by avoiding collisions of the directional DATA (DDATA) and DACK packets. The omnidirectional RTS (ORTS) mechanism is used to mitigate deafness when directional antennas are in the unblocking state. But, directional antennas overheard RTS and CTS are blocked and node having one or more blocked directional antennas uses DRTS. Therefore, the DMAC protocol cannot prevent deafness caused by directional transmission.

[*] This research was supported by the MIC(Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Assessment)" (IITA-2005-C1090-0502-0009).

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 594 – 602, 2006.

The dual-channel ad hoc MAC protocols [8][9] have been proposed to improve spatial reuse by using the separated channels. In the dual-channel MAC (DUCHA) protocol [9], RTS and CTS are transmitted over control channel and DATA and ACK are transmitted over data channel. The negative CTS (NCTS) and an out-of-band busy tone are used to prevent collisions of DATA and ACK in data channel. However, omnidirectional antenna in DUCHA degrades spatial reuse. In this paper, we propose a dual-channel ad hoc MAC protocol with ORTS, OCTS, DDATA, and DACK. In the proposed MAC protocol, the ORTS and OCTS mechanisms in control channel overcome deafness and the DDATA and DACK mechanisms in data channel improve spatial reuse by using the efficient blocking algorithm for directional antennas. The throughput performance of the proposed MAC protocol is confirmed by using Qualnet ver. 3.8 simulator [10]. The rest of this paper is organized as follows: In Section Ⅱ , we describe the DMAC and DUCHA protocols. We describe the operation of the proposed MAC protocol in section Ⅲ. The simulation results are presented in Section Ⅳ. Finally, we conclude this paper in Section Ⅴ.

2 Related Work

2.1 Directional MAC Protocol

The DMAC protocol [2] with an omnidirectional antenna and directional antennas has been proposed to improve spatial reuse. In the DMAC protocol, ORTS is used when directional antennas are unblocked and DRTS is used when one or more directional antennas are blocked. The directional antennas overheard RTS or CTS are blocked to prevent collisions of the DDATA and DACK packets. Figure 1 shows the operation of the DMAC protocol. Node A having the unblocked directional antennas transmits ORTS to node B and node B sends OCTS to node A. The DATA and ACK packets are transmitted by directional antennas. Node C in the coverage of node A overhears ORTS of node A and the directional antenna is blocked. Node C having the blocked directional antenna sends DRTS to node D and therefore, directional transmission can improve spatial reuse. Node E sends DRTS to node C because Node E does not overhear DRTS of node C. Node E retransmits DRTS to node C in deafness after a backoff period. The backoff period of node E is exponentially increased at every retransmission and the throughput performance is degraded. Therefore, directional transmission has trade-off between spatial reuse and deafness.

2.2 Dual-Channel MAC Protocol

The dual-channel MAC (DUCHA) protocol [9] separates control channel and data channel and uses an out-of-band busy tone as shown in figure 2. In the DUCHA protocol, omnidirectional transmission solves the problem of deafness and an out-of-band busy tone and negative CTS (NCTS) decreases the hidden/exposed terminal problems. In the NCTS mechanism, the node transmits negative CTS (NCTS) over data channel instead of CTS to the transmitter and prevents collisions of DATA and ACK in data channel. In the DUCHA protocol, the receiving node transmits an out-of-band busy tone during the reception of DATA. If DATA is correctly received the node stops the busy tone and terminates the communication. If the reception of DATA is failed the negative ACK (NACK) signal of the continued busy tone for an appropriate period is transmitted to the transmitting node. When NACK is sensed during the NACK period, the node starts the retransmission procedure. Therefore, an out-of-band busy tone solves the hidden terminal problem.

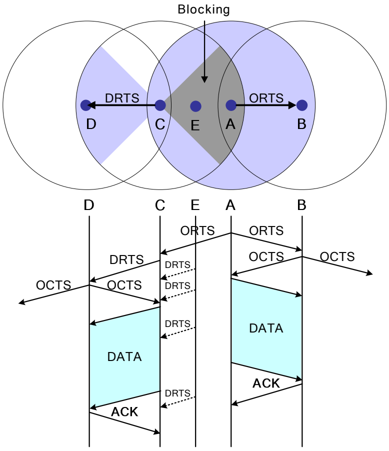


Fig. 1. The DMAC Protocol with ORTS or DRTS: The circle centered at each node shows its transmission range. An arrow represents the transmission direction. The flow chart shows the message flow between the communication nodes.

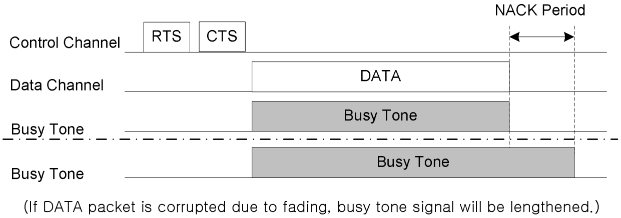
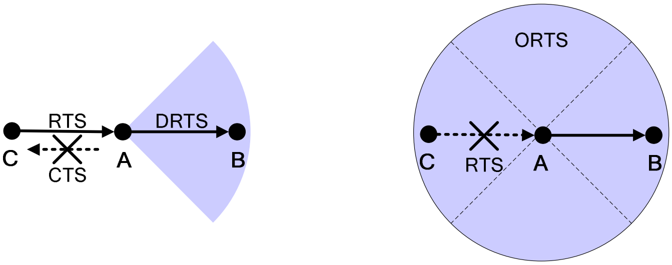


Fig. 2. The DUCHA Protocol with the separated control channel and data channel

3 A Dual-Channel MAC Protocol with Directional Antennas

In this paper, we propose a location aware dual-channel MAC protocol with ORTS, OCTS, DDATA, and DACK. The location information of nodes can be obtained by using the global positioning system (GPS) receiver. The ORTS and OCTS packets are transmitted over control channel and the DDATA and DACK packets are transmitted over data channel. Figure 3 shows that omnidirectional transmission in control channel can overcome deafness. In figure 3(a), node transmits DRTS to node B and therefore, node C does not overhear RTS of node A. Node C transmits RTS to node A but node A does not hear and respond to RTS of node C. Node A gets into deafness and node C retransmits RTS to node A. The backoff period of node C is exponentially increased at every retransmission. In figure 3(b), node A sends ORTS over control channel to node B and node C overhears RTS of node A. Node C stores the information of node A into the deafness table. Node C checks the destination node of node A in the deafness table before the RTS transmission and does not transmits RTS to node A in communication. The omnidirectional transmission and deafness table solve the problem of deafness.

Figure 4 show the blocking problem for directional antennas. In the DMAC protocol, nodes C and D overhear ORTS or OCTS of node A and directional antennas overheard RTS or CTS are blocked to prevent collisions of ORTS/OCTS and DDATA/DACK transmitted over the same channel. However, the separated channels in the proposed MAC protocol avoid collisions of ORTS/OCTS and DDATA and DACK as shown in figure 4(b). In the proposed MAC protocol, the new blocking algorithm for directional antennas used in data channel should be required. The NCTS mechanism as that in DUCHA is used to solve the hidden terminal problem.



(a) DRTS in the DMAC protocol (b) ORTS in the proposed scheme

Fig. 3. The ORTS mechanism in the proposed MAC protocol mitigates deafness

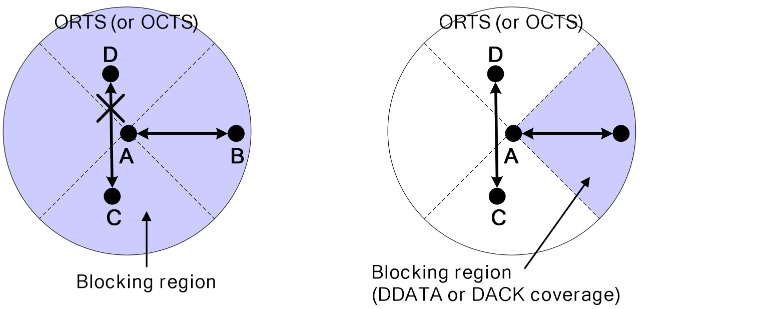
3.1 A Blocking Algorithm for Directional Antennas

Figure 5 shows the blocking region for directional antennas overheard ORTS or OCTS. The nodes C and B overheard ORTS or OCTS of node A determines the blocking region as follows:

① Node A transmits RTS (or CTS) with the overhead information of the main angle ϕ of the directional antenna as shown in figure 5. If the node has 4 directional antennas, the blocking region is in the range of (ϕ-45, ϕ+45)

② Nodes B and C calculate the angles θ_1 and θ_2 with respect to north as shown in figure 5.

③ The directional antenna of node C is blocked as the angle θ_2 is in the blocking region and the directional antenna of node B is unblocked as θ_1 is not in the blocking region.



(a) Blocking region in the DMAC protocol (b) Blocking region in the proposed protocol

Fig. 4. The proposed MAC protocol improves spatial reuse using the new blocking algorithm for directional antennas

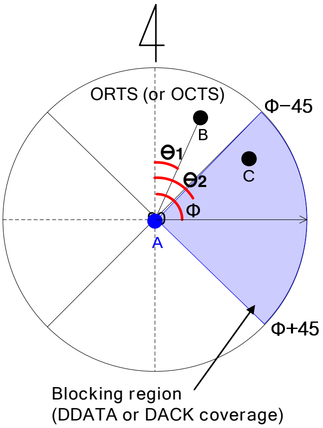


Fig. 5. The nodes B and C overheard RTS/CTS determine the blocked directional antennas. Node C in the blocking region of node A blocks the directional antenna overheard RTS. But directional antennas of node B are in the unblocking states.

3.2 Deafness Table

The node overheard ORTS and OCTS on control channel stores the overheard information into the deafness table as shown in figure 6. For example, nodes C and D in figure 4(b) store the information of node A in their deafness tables. The deafness table is composed of a source address, a destination address, a frame type, and the received time of the overheard RTS or CTS. Node checks the deafness table before the RTS transmission whether the destination node is in the deafness table or not. If the destination node is in the deafness table, the node starts a deafness timer and waits for the expiration of that timer. When the timer is expired node eliminates the information of the destination node in the deafness table and starts the transmission procedure.

Destination Address	Source Address	Frame Type	Receive Time

Fig. 6. Deafness table format

4 Simulation Results and Discussions

In this chapter, we confirm the throughput performance of the proposed MAC protocol by computer simulations using Qualnet ver. 3.8 simulator. Simulation scenario is composed of a single-hop random topology of 60 nodes and a 5-hop mesh topology of 36 nodes. The average throughput of the proposed MAC protocol is compared with the DMAC, DUCHA, and IEEE 802.11 MAC protocols.

4.1 Simulation Environments

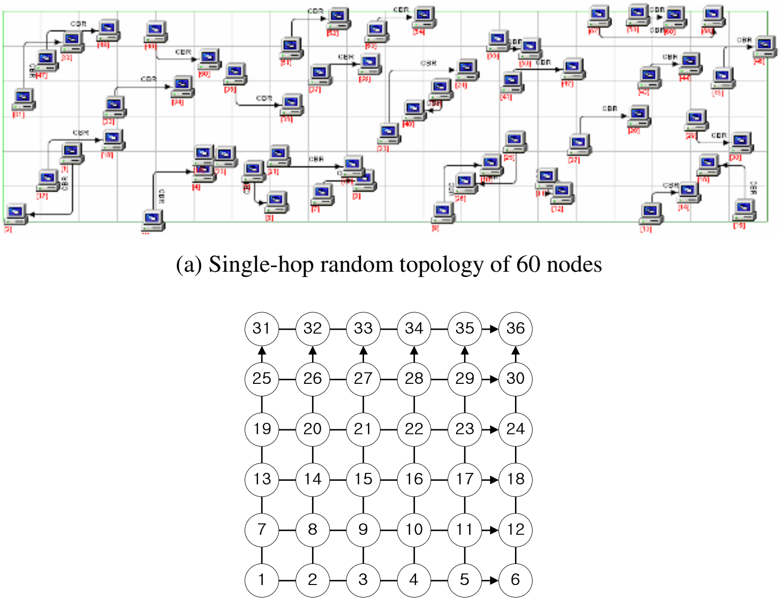
We use the physical layer of the IEEE 802.11b standard and data rate of 2Mbps. In the proposed MAC protocol and the DUCHA protocol, data rate of control channel and data channel are the same of 1Mbps and transmission range is approximately 250m. We use a static routing, constant bit rate (CBR) traffic, the size of the DATA packet of 1000byte, and 8 switched beam antennas. The important simulation parameter values are shown in table 1. Figure 7(a) shows a single-hop random topology of 60 nodes. In this scenario, 60 nodes are randomly arranged into the rectangle area of 300x1000m^2. The destination nodes are randomly selected in the transmission range of 250m. Figure 7(b) shows a 5-hop mesh scenario of 36 nodes. This scenario arranges 36 nodes into the square area of 1500x1500m^2 and the distance of each node is 240m as shown in figure 7(b).

Table 1. Default values used in the computer simulations

	Random topology	**Mesh topology**
CBR traffic	0.2 ~ 1.0**Mbps**	0.1 ~ 0.5**Mbps**
Distance between nodes	0~250m(random)	240m(fixed)
Data Rate : 2Mbps	**Control channel**	**Data channel**
	1Mbps	1Mbps
Transmission range	250m	
DATA packet size	1000byte	
Simulation time	120sec	

4.2 Simulation Results

The average throughput performance of the proposed MAC protocol in a single-hop random topology and a 5-hop mesh topology are shown in figures 8 and 9, respectively. Figure 8 shows the average throughput of the proposed MAC protocol in a single-hop random topology. The average throughputs of the proposed MAC protocol



(a) Single-hop random topology of 60 nodes

(b) 5-hop 6x6 mesh topology

Fig. 7. Ad hoc network topologies for computer simulations

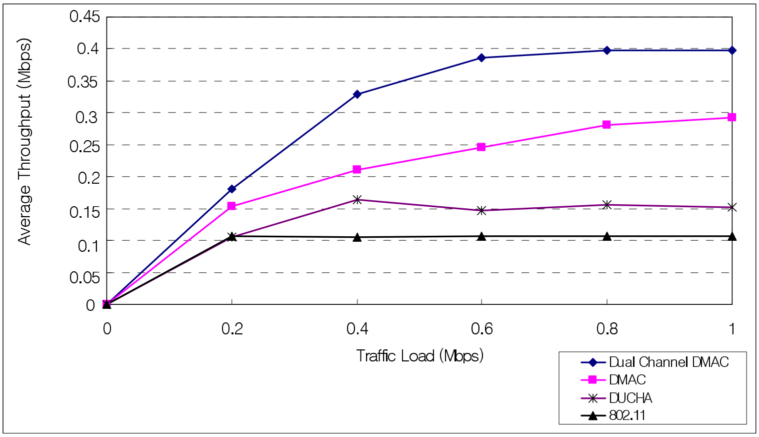


Fig. 8. The average throughput performance of the proposed MAC protocol in a single-hop random topology

and DMAC with directional antennas are better than those of IEEE 802.11 MAC protocol and DUCHA with an omnidirectional antenna. In the proposed MAC protoccol, the separate channels overcome deafness and the efficient blocking algorithm for directional antennas used in data channel improves reuse. For these reasons, the throughput performance of the proposed MAC protocol is better than that of the DMAC protocol. In figure 8, the average throughputs are 398.0kbps, 292.1kbps, 151.8kbps, 107.1kbps for the proposed MAC protocol, DMAC, DUCHA, IEEE 802.11 MAC protocols at the traffic load of 1Mbps, respectively.

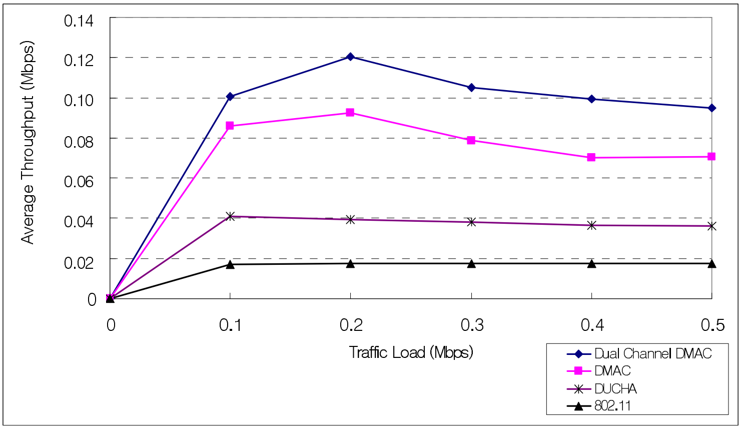


Fig. 9. The average throughput of the proposed MAC protocol in a 5-hop mesh topology

Figure 9 shows that throughput performance of the proposed MAC protocol in a 5-hop 6x6 mesh topology. The average throughputs of the proposed MAC protocol and DMAC with directional antennas are superior to those of IEEE 802.11 MAC and DUCHA protocols with an omnidirectional antenna in the multi-hop mesh topology. In the multi-hop environments, deafness caused by the directional transmission degrades the through performance of ad hoc networks. The omnidirectional transmission on control channel and the deafness table improve the average throughput of the proposed MAC protocol. The average throughput of the proposed protocol is superior to that of the DMAC protocol in the multi-hop environments. The average throughputs are 95.0kbps, 70.5kbps, 36.2kbps, and 17.3kbps for the proposed MAC protocol, DMAC, DUCHA, and IEEE 802.11 MAC protocols at the traffic load of 0.5Mbps, respectively.

5 Conclusions

In this paper, we propose the dual-channel ad hoc MAC protocol with ORTS/OCTS on control channel and DDATA/DACK on data channel. In the proposed MAC protocol, ORTS/OCTS and the deafness table overcome deafness caused by directional transmissions. We also propose the blocking algorithm for directional antennas based on the overhead location information in ORTS and OCTS. That improves spatial reuse and therefore, the proposed ad hoc MAC protocol has a good throughput performance. We confirm the throughput performance of the proposed MAC protocol in a single-hop topology of 60 nodes and a 5-hop 6x6 mesh topology by using Qualnet ver 3.8 simulator. Simulation results show that the average throughput of the proposed MAC protocol is better than those of the DMAC, DUCHA, and IEEE 802.11 MAC protocols in the single-hop and multi-hop environments.

References

1. Murthy Manoj, Ad Hoc Wireless Networks – Architectures and Protocols, Prentice Hall, (2004).
2. Y. Ko, V. Shankarkumar, and N. H. Vaidya, "Medium Access Control Protocols Using Directional Antennas in Ad Hoc Networks," IEEE INFOCOM 2000, March 2000, pp. 13-21.
3. R. R. Choudhury, X. Yang, N. H. Vaidya and R. Ramanathan, "Using directional antennas for medium access control in ad hoc networks," MOBICOM(2002) 57-90.
4. L. Bao and J. Garcia-Luna-Aceves, "Transmission scheduling in Ad hoc networks with directional antennas;" " MOBICOM(2002) 48-58.
5. S. Yi, Y. Pei, and S. Kalyanaraman, "On the capacity improvement of Ad hoc wireless networks using directional antennas," Mobihoc(2003) 108-116.
6. M. Sanchez, T. Giles, and J. Zander, "CSMA/CA with beam forming antennas in multihop packet radio," Proceeding of Swedish Workshop on Wireless Ad Hoc Networks(2001) 63-69.
7. T. Korakis, G. Jakllari, and L. Tassiulas, "A MAC protocol for full exploitation of directional antennas in Ad hoc wireless networks," Mobihoc(2003) 98-107.
8. A. Muqattash and M. Krunz, "Power Controlled Dual-channel (PCDC) Medium Access Protocol for Wireless Ad hoc Networks," IEEE INFOCOM(2003) 470-480.
9. H. Zhai, J. Wang, Y. Fang, and D. Wu, "A Dual-channel MAC Protocol for Mobile Ad Hoc Networks," IEEE Workshop on Wireless Ad Hoc and Sensor Networks, in conjunction with IEEE Globecom(2004) 27-32.
10. Scalable Network Technologies, Qualnet simulator version 3.8,
www.scalable-networks.com.

A Power-Efficient Design Employing an Extreme Condition Detector for Embedded Systems*

Hyukjun Oh[1], Heejune Ahn[2], and Jiman Hong[1,**]

[1] Kwangwoon University, Seoul, Korea
{hj_oh, gman}@kw.ac.kr
[2] Seoul National University of Technology, Seoul, Korea
heejune@snut.ac.kr

Abstract. In this paper, a power-efficient scheme for embedded systems with wireless communication applications is proposed to reduce the power consumption of the overall system. Any transmission related module is required to be on only when the transmission is active and reliable to reduce power dissipation. The proposed method is based on the use of the extreme channel condition detector that is designed to detect the extremely bad channel condition. Under such a condition, carrying user information over the air link is completely impossible. The considerable power reduction is achieved by turning off several modules within the embedded system related to the information transmission like LCD, image encoder, voice encoder, and power amplifier under this condition. Moreover, a simple extreme channel condition detector is also proposed in this paper. The design example on the selected platform shows that the proposed scheme is very efficient in power saving for the embedded system.

1 Introduction

An embedded system is a specialized computer system that is a part of a larger application specific system or machine. Typically, an embedded system is housed on a single microprocessor board with the programs stored in ROM. Some embedded systems include an operating system, but many are so specialized that the entire logic can be implemented as a single program [1]. They are characterized by the presence of a dedicated processor for that specific application software. Recently, there have been remarkable growths of such systems in many sectors of markets. Most of popular smart electronics around us are such embedded systems nowadays and the number of such smart electronics and the demand for them is rapidly increasing continuously. A notable example is cellular phones in which dedicated embedded processors control each aspect of the power efficiency, quality of services and various different applications of the

* The present research has been conducted by the Research Grant of Kwangwoon University in 2005.

** Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 603–611, 2006.

phones. The major factor of leading to their growths is the increasing complexity of such smart electronics, which accelerate migration from application specific "logic" to application specific "code" running on the existing dedicated processors.

Minimizing power and energy dissipation is a key factor in embedded system designs. Peak power consumption is related to power supply design for the system. On the other hand, average power consumption is directly related to battery life, hence it may be the critical factor in the portable embedded systems [2]. The design of embedded systems with less power dissipation is a challenging task for today's design environments. As opposed to a general-purpose system, an embedded system performs just one particular and specific application. Therefore, the system should be designed with respect to the particular application to have lower cost, higher performance, or be more power-efficient. As mobile systems become more popular and popular, how to length the battery life of these systems becomes a critical issue, especially in the embedded system with applications of wireless communications. This has led to a significant research effort in power-efficient designs/modelings [2]-[9]. It is natural to think of ways to reduce the power consumption considering and utilizing inherent properties of the wireless communications. The use of such properties requires power-efficient design from the inside of the communication module, and it will provide satisfactory power reduction performance inherently. However, most previous works to date have treated it as a given and untouchable module and have focused on the power reduction from the top of it [2]-[6].

In this paper, a power-efficient scheme for embedded systems with wireless communication applications is proposed to reduce the power consumption of the overall system. Any transmission related module is required to be on in such applications only when the transmission is active and reliable to reduce power dissipation. The proposed method is based on the use of a signal processing algorithm for the extreme channel condition detection that is designed to detect the extremely bad channel condition. Under such a condition, carrying user information over the air link is completely impossible. The considerable power reduction is achieved by turning off several modules within the embedded system related to the information transmission like LCD, image encoder, voice encoder, and power amplifier under this condition. A simple software code can serve as a brain of such tasks given primitive signals from the detector. Moreover, a simple extreme channel condition detector is also proposed in this paper. The design example on the selected platform shows that the proposed scheme is very efficient in power saving for the embedded system.

This paper is structured as follows: Section 2 describes our embedded system model with the applications of wireless communication. In Section 3, we present the proposed power reduction scheme using the indicator of extremely bad wireless channel condition. Section 4 addresses the proposed signal processing algorithm to detect the worst propagation channel condition used in Section 3. An example is presented to demonstrate the efficiency of the proposed method in Section 5.

2 System Model

In this section, the target embedded system model is addressed. A usual embedded system is shown in Fig. 1, which comprises a processor core, an instruction cache, a data cache, a main memory, and a custom hardware part (ASIC/FPGA) [5]. Once we assume that hardware/software partitioning has been already performed and the custom hardware is fixed, it is regarded as a simple block adding a constant amount of power dissipation to the system model. Then, most previous works have tried to minimize power consumption from the perspective of the rest blocks [2]-[6]. It leads to instruction-level power optimization [7]. The custom hardware part, actually, means any module or part that we cannot do anything to reduce power consumption in the previous works. Basically, these approaches regard the custom hardware part as an untouchable block and it is not considered in the power optimization efforts. Usually, the communication module in the system with the wireless communication application has been considered as one of this kind. In general, such an assumption is valid and useful when the application of the target system is not for wireless communications. In the embedded system with the application of the wireless communication, considerable portion of total power consumption is caused by the transmission related functionality itself in that module. Therefore, it is desirable to include the communication module in optimizing the power consumption of the overall system. The details are discussed in Section 3.

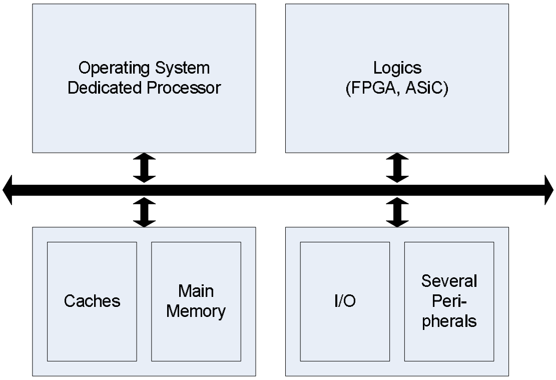


Fig. 1. A usual embedded system model

Because our interest is focused on the system with the application of wireless communications, the best target system would be cellular phone. Fig. 2 shows simplified block diagram of a cellular phone. It shows only major components including dedicated processors. Mainly, it consists of three parts: application related components like several multimedia peripherals and application processors, modem related components including baseband processors, and RF chains. It is assumed that the dedicated application or modem processor is equipped with an operating system and it serves as a brain for performing the proposed power management scheme in this paper. A real time operating system (RTOS) would be appropriate for this purpose, but we do not limit the form of possible operating system in our study, because the operating system is not of our interest. It simply provides room for central control of the power management scheme. The embedded processors currently used in designs take two possible shapes: microprocessors or digital signal processors (DSP's). They can be used separately as a single component, or they can be incorporated in a larger silicon chip in the form of embedded cores along with program/data memory and other dedicated logic. In our system model, any form and any kind of dedicated processors are allowed. They are just target processors to load the operating system.

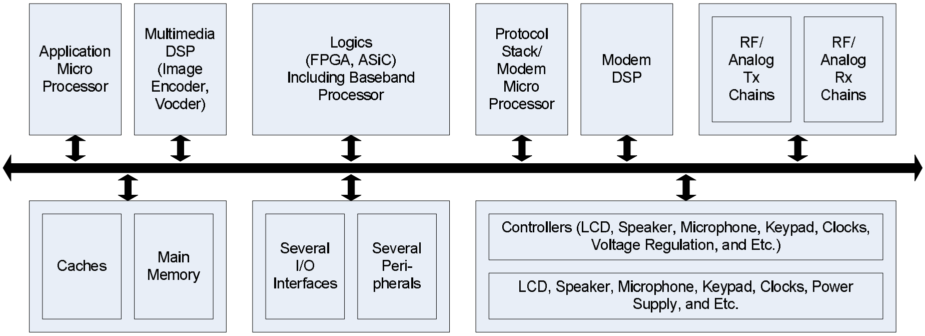

Application Micro Processor
Multimedia DSP (Image Encoder, Vocder)
Logics (FPGA, ASiC) Including Baseband Processor
Protocol Stack/ Modem Micro Processor
Modem DSP
RF/ Analog Tx Chains
RF/ Analog Rx Chains
Caches
Main Memory
Several I/O Interfaces
Several Peri-pherals
Controllers (LCD, Speaker, Microphone, Keypad, Clocks, Voltage Regulation, and Etc.)
LCD, Speaker, Microphone, Keypad, Clocks, Power Supply, and Etc.

Fig. 2. A system model of a cellular phone

3 Proposed Power Management Scheme

An embedded system is specialized for one particular application that is known a priori. Therefore, the system can be designed with respect to the particular application to have low power consumption. In this paper, we consider the system with the application of wireless communication. Then, it is worthwhile to take inherent properties of the particular application of the wireless communication as beneficial resources to reduce the power dissipations.

Its "wireless" propagation channel characterizes the wireless communication. Its applications are inevitably closely related to the propagation channel and channel conditions. Considerable power in such a system is dissipated to perform basic functionality of the transmission over the channel. In normal channel condition, there is nothing special to consider for the purpose of power dissipation reduction from the view of the overall system. The power is well managed by the modem related functionality itself in the basedband processor. In this case, several existing system models are useful and many previous works based on them can be applied. That is, any power saving schemes assuming the transmission related block as a simple black-box with constant power dissipation can be used in general. However, it is not valid anymore in unsatisfactory channel condition. In the extremely bad channel condition, the real time transmission over the unreliable air link is not possible. It means that operating any transmission related code, function, module, logic, and component in the system is meaningless. The situations get worse as the system is operated around cellular service boundaries because the portion of power consumption for transmission related operation get bigger. Therefore, it is necessary to design the power saving scheme considering the overall system including the transmission related block that has been considered as a black-box before.

Such a channel condition mentioned above can happen often in wireless environment due to the inherent characteristic of mobile channels of wide and fast variations. In the case of channel being unrecoverable or in the case of such a bad channel condition lasting long enough, a dedicated higher layer function in the protocol stack kicks in to control radio link in most popular wireless communication systems [10]. However, the most trouble case is when the propagation channel falls in and out such an extremely bad conditions quite often or continuously. It is not serious and long enough to be regarded as an unrecoverable case where the dedicated higher layer function kicks in. One example for this case is when the system is running in the vehicle passing tunnels.

In this paper, we propose to disable or to turn off the meaningless codes, functions, modules, logics, and components to minimize the power consumption when the extreme channel condition is detected. We use the term of "components" for any software and hardware unit that can be disabled or turned off separately throughout the paper. Some components do not need to be operated in full performance under such an extreme case for power-efficient operations. The operating system on the dedicated processor serves as a central control unit of managing and performing the required operations for the incoming extreme channel condition indicator from the signal processing unit in the modem. First, we need to identify which components can be disabled under this situation. Those components are dependent on the configuration of the target embedded system. In our system model as in Section 2, LCD, image encoder, vocoder, and RF chains including power amplifier can be turned off. Furthermore, the data memory can be flushed and/or the number of accessing internal or external memory can be minimized. Such a serise of opearions can be regarded as another sleep mode that consumes more power than the real sleep operation. Note that RF chain is turned off only during user data transmission. It should be turned on again during the pilot transmission to get recovered in the forthcoming normal channel condition. Several variants are also possible from the proposed method as per the architecture of the target embedded system. For example, the user can feel uncomfortable when LCD is turned off completely in this situation. Instead, alternative variants like diminishing the brightness of LCD's or making a low power LED blinking can be used. The proposed power-efficient scheme is summarized in Fig. 3. The performance of the proposed scheme strongly depends on the signal processing algorithm developed to detect the extremely bad channel condition. The signal processing algorithm of the detector is presented in Section 4.

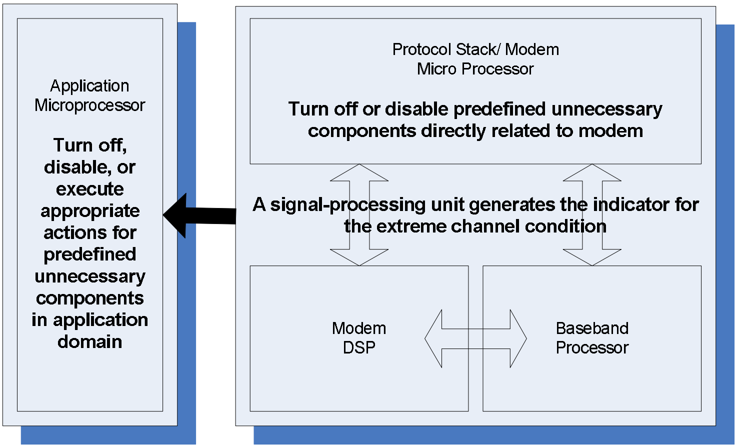

Application Microprocessor
Turn off, disable, or execute appropriate actions for predefined unnecessary components in application domain
Protocol Stack/ Modem Micro Processor
Turn off or disable predefined unnecessary components directly related to modem
A signal-processing unit generates the indicator for the extreme channel condition
Modem DSP
Baseband Processor

Fig. 3. The proposed power saving scheme using the extreme channel state indicator

4 Detection of Extreme Channel Condition

A computationally efficient signal-processing algorithm is proposed to detect the extremely unstable channel condition. The algorithm should be simple to implement so that it does not add another noticeable power dissipation to the overall system. The proposed algorithm is based on the use of channel state information (CSI) that is available in most wireless communication system. CSI's are available in the form of the signals from the base stations or in the form of direct estimations by the end-user equipment. Any form of CSI's can be used for this purpose.

In this section, we consider the CDMA cellular systems that are the most widely accepted third generation standards for the cellular communications in the world [10]. However, the proposed scheme is not limited to the applications with the CDMA systems. It can be generalized easily to other systems like OFDM based ones. In CDMA systems, several different CSI's are estimated or transmitted by the end-user terminal. The CSI's of the transmitted power control (TPC) bits are available in plenty of time in CDMA systems 3GPP. They are binary state information. One is indicating that the current channel quality is good enough to satisfy the required quality of service. The other is representing the channel condition that is not good enough to achieve the reliable transmissions. Hence, the latter is indicating the channel condition that we need to detect. The proposed detection scheme simply estimates the frequency of occurrence of this unsatisfactory channel state by counting the number of the second state of the TPC bits during the given time period. Then, the current channel condition is regarded as the extremely bad state if the majority of the TPC bits are the second state in the given time frame. That is, if the counted number of the occurrence of the corresponding CSI in the fixed time duration is larger than a threshold, the channel condition is not adequate for the reliable transmissions anymore.

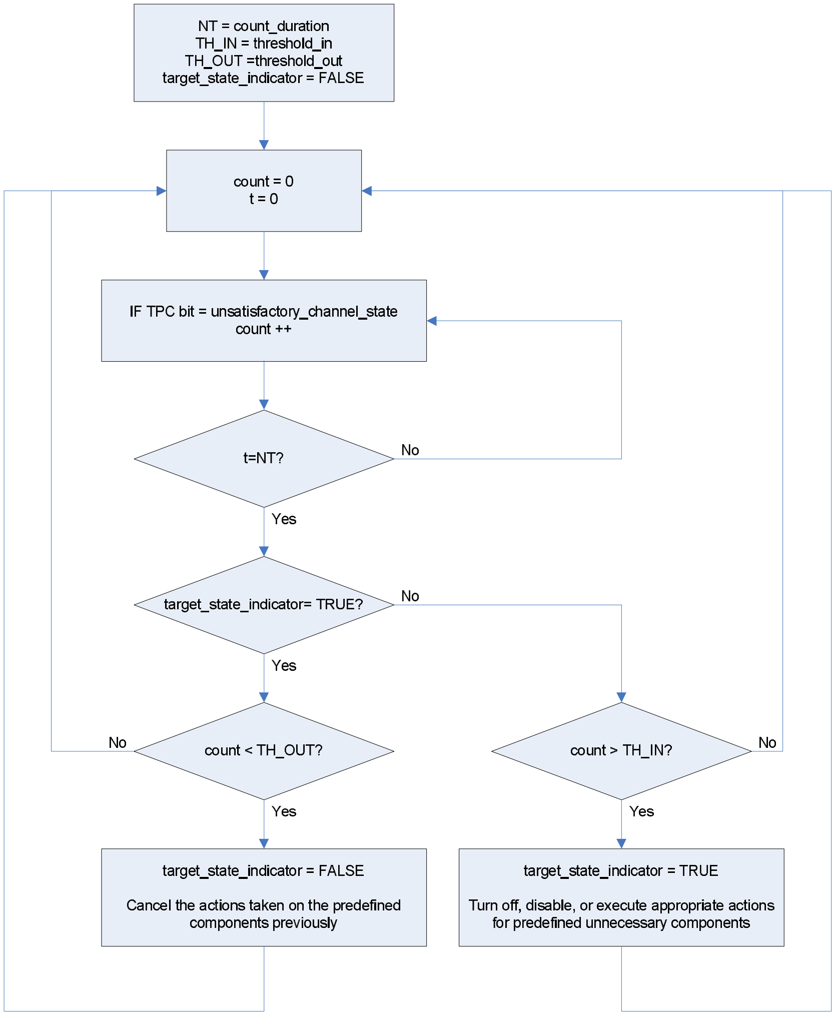

NT = count_duration
TH_IN = threshold_in
TH_OUT =threshold_out
target_state_indicator = FALSE
count = 0
t = 0
IF TPC bit = unsatisfactory_channel_state
count ++
t=NT?
No
Yes
target_state_indicator= TRUE?
No
Yes
count < TH_OUT?
No
Yes
count > TH_IN?
No
Yes
target_state_indicator = FALSE
Cancel the actions taken on the predefined components previously
target_state_indicator = TRUE
Turn off, disable, or execute appropriate actions for predefined unnecessary components

Fig. 4. The overall flow chart of the proposed scheme along with the signal-processing algorithm

The details of the algorithm are shown in Fig. 4. Two different values of thresholds are used to introduce the hysteresis in the detection. The different thresholds are used in each transition of the channel state falling in and out from the unsatisfactory state. This scheme is efficient in the severe time variant system. The proposed signal processing algorithm can be implemented in either logics or codes. Fig. 4 also summarizes the overall power management algorithm along with the proposed power management scheme in Section 3.

5 Simulation Results

The proposed power management scheme based on the detection of the extreme channel condition is applied to the system model in Section 2. The communication related functions are modeled based on UMTS specifications [10] and practical factors like fading, shadowing, and path loss are considered in the channel model. The path loss exponent of four is used. Multiple 6 to 7 minute (about 20000 TTI's) calls are simulated. In our simulation, the LCD block is excluded from the final power saving ratio computation becuase it is usually automatically dimmed out during the call.

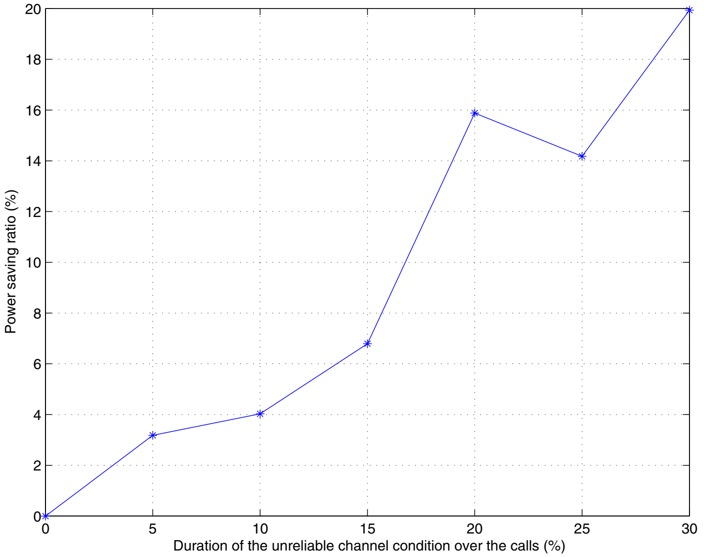


Fig. 5. Obtained power savings with the proposed method

The considerable power saving is achieved employing the proposed method as showin Fig. 5. The amount of power saving is increased as the duration of the unreliable channel condition gets longer as expected. In fact, the performance depends on how long and how often the extreme channel condition occurs. The fluctuating curve shows the effect of noisy extreme channel condition detectors. It is a possible further work to improve the performance of the detector while the overall complexity is retained. If the duration of such a state is too long, a higher layer radio link control function dedicated to deal with this situation kicks in and it takes power management control over the proposed scheme.

6 Conclusions

In this paper, a possible way to reduce the power dissipation considering the overall system including the communication module has been proposed. We have discussed a power-efficient scheme by turning off or disabling unnecessary components under the unreliable channel condition in the embedded system with the application of the wireless communication. A signal processing algorithm to detect such a channel condition is also proposed using the channel state information available in most cellular communication systems. Throughout the simulation, we have shown that our proposed method can save power considerably in the case of the channel getting in and out of the extreme state continuously.

References

1. http://www.webopedia.com/TERM/E/embedded_system.html
2. Simunic, T., Micheli, G., Benini, L.: Energy efficient design of battery-powered embedded systems. Proc. Int. Symp. Low Power Electronics and Design (1999) 212–217
3. Benini, L., Bogliolo, A., Micheli G.: A survey of design techniques for system-level dynamic power management. IEEE Trans. Very Large Scale Integration systems, Vol. 8 (2000) 299-316
4. Benini, L., Micheli, G.: System-Level Power Optimization: Techniques and Tools ACM Trans. Design Automation of Electronic Systems, Vol. 5 (2000) 115-192
5. Li, Y., Henkel, J.: A framework for estimating and minimizing energy dissipation of embedded HW/SW systems. Proc. Int. Conf. Design Automation (1998) 188–193
6. Catthoor, F., Franssen, F., Wuytack, S., Nachtergaele, L., Man, H.: Global communication and memory optimizing transformations for low power signal processing systems. Proc. Int. Wkshp. on Low Power Design (1994) 203–208
7. Tiwari, V., Malik, S., Wolfe, A.: Power analysis of embedded software: A first step towards software power minimization. IEEE Trans. VLSI Systems (1994) 437–445
8. Hsieh, C., Pedram, M., Mehta, G., Rastgar, F.: Profile-driven program synthesis for evaluation of system power dissipation. Proc. Design Automation Conf. (1997) 576–581
9. Simunic, T., Benini, L., Micheli, G.: Cycle accurate simulation of energy consumption in embedded systems. Proc. Design Automation Conf. (1999) 867–872
10. 3GPP Specifications, Release 5.

An Efficient Delay Metric on RC Interconnects Under Saturated Ramp Inputs

Ki-Young Kim[1], Seung-Yong Kim[1], and Seok-Yoon Kim[2]

[1] Department of Computer, Soongsil University,
1-1 Sang-Do 5 Dong, Dong-Jak Gu,
Seoul, Korea
{kky, seeon}@ic.ssu.ac.kr

[2] Department of Computer, Soongsil University,
1-1 Sang-Do 5 Dong, Dong-Jak Gu,
Seoul, Korea
ksy@comp.ssu.ac.kr

Abstract. This paper presents a simple and fast delay metric RC-class interconnects under step and saturated ramp inputs. The proposed RC delay metric under step input, called MECM(Modified ECM), provides a reasonable accuracy without using circuit moments. The next RC delay metric under saturated ramp inputs, called FDM(Fast Delay Metric), can estimate delay times at an arbitrary node using a simple closed-form expression and is extended from MECM easily. As compared with similar techniques proposed in previous researches, it is shown that the FDM technique involves much lower computational complexity for a similar accuracy. As the number of circuit nodes increases, there will be a significant difference in estimation times of RC delay between the previous techniques based on two circuit moments and the FDM which do not depend on circuit moments.

1 Introduction

As circuit technology goes beyond the ultradeep submicrometer regime, interconnect delays increasingly dominate gate delays. That is, the accurate estimation of interconnect delays becomes significantly important in designing high-speed and high-density systems. Therefore designers have to verify timing with great attention in the early process of design optimization such as place route, floorplanning, interconnect sizing, buffer insertion, etc. Efficient delay metric for estimating interconnect delays, which may be computed millions of times during the inner process of design optimization, is a critical issue.

Studies about estimation of delay times of RC-class interconnects under step input have been presented by many papers till now. Those studies include the method proposed by Kahng and Muddu [1], the PRIMO proposed by Kay and Pileggi [2], the h-gamma proposed by Lin and Acar [3] and the D2M and ECM proposed by Alpert and Devgan [4]. Since the above methods depend on circuit moments, however, they are not the most efficient solution. An effective closed-form delay metric would be a better choice if a reasonable accuracy is sustained.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 612–621, 2006.

This paper proposes a new closed-form RC delay metric without using circuit moments.

Moreover, since interconnects are driven by nonlinear devices(gates), it is necessary to model the input signal of interconnects as a finite saturated ramp input. However, studies about the estimation of delay times under saturated ramp inputs have not been performed as actively as those for the step input case. The previous studies include the method proposed by Menezes and Pullela [5], the method proposed by Kahng and Muddu [6], however, those delay metrics are not in the most simple form. A recent study [7] has proposed the PERI(Probability distribution function Extension for Ramp Inputs) technique that extends delay metric for step inputs to ramp inputs. The PERI is the most efficient method among the previous studies although it still requires two circuit moments. This paper proposes the FDM(Fast Delay Metric), that extends delay metric for step inputs to ramp inputs without using circuit moments. As the number of circuit nodes increases, the time to compute circuit moments grows more, hence the timing analysis based on FDM will be much faster than that based on PERI.

This remainder of the paper is organized as follows. Section 2 proposes a new RC delay metric without using circuit moments for the step input case. Section 3 describes a new technique to extend delay metric for step input to ramp input case. Section 4 shows experimental results, and a conclusion is given in Section 5.

2 A New RC Delay Metric Under Step Input

A new RC delay metric proposed in this section is based on the ECM(Effective Capacitance Metric) concept proposed in [4]. As the model used in the ECM is synthesized using the method proposed by O'Brien and Savarino [8], it requires calculating circuit moments. We propose the MECM(Modified ECM), which does not depend on circuit moments while keeping the accuracy.

2.1 ECM

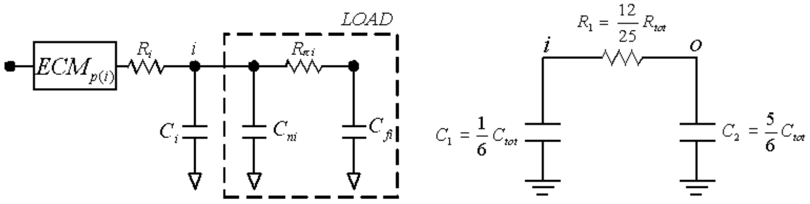
To compute delay time at node i, the ECM transforms total load at node i into a model and then replaces the model by a single capacitance based on the effective capacitance concept [9], as shown in Figure 1(a).

Defining that the single capacitance replacing the total load is C_{eff} and delay time at the previous node of i is $ECM_{p(i)}$, delay time at node i, ECM_i is given in Eq.(1).

$$ECM_i = ECM_{p(i)} + R_i(C_i + C_{eff}) \tag{1}$$

2.2 MECM

MECM(Modified ECM) improves the drawback of ECM using circuit moments when it transforms total load at node i into a model. We can synthesize a model using only the total resistance(R_{tot}) and capacitance(C_{tot}) values of load at node i [10], as shown Figure 1(b). That is, we may use characteristic data


LOAD
$ECM_{p(i)}$
R_i
i
$R_{\pi i}$
C_i
C_{ni}
C_{fi}
$R_1 = \frac{12}{25} R_{tot}$
i
o
$C_1 = \frac{1}{6} C_{tot}$
$C_2 = \frac{5}{6} C_{tot}$

(a) (b)

Fig. 1. (a)ECM, (b)π model proposed by Kahng

of interconnects to synthesize the π model. Using this π model, MECM can transform the total load into a reduced model based not on circuit moments.

The other feature of MECM is that it adds ground capacitance at node i, C_i, to the total load before we synthesize π model to compute delay time at node i. By doing this, the estimation of delay time becomes more accurate. Figure 2 shows the concept of MECM.

Using R_{tot}, C_{tot} and the Elmore Delay at node i, T_{ED}, delay time at node i, $MECM_i$, can be derived as in Eq.(2).

$$\begin{aligned} MECM_i &= MECM_{p(i)} + R_i C_{eff} \\ &= MECM_{p(i)} + R_i(C_1 + C_2(1 - e^{\frac{-T_{ED}}{R_1 C_2}})) \\ &= MECM_{p(i)} + \frac{R_i C_{tot}}{6}(6 - 5e^{\frac{-5T_{ED}}{2R_{tot}C_{tot}}}) \end{aligned} \tag{2}$$

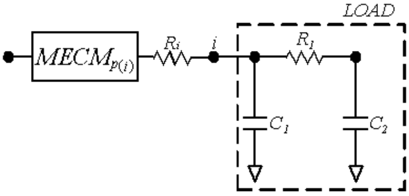


Fig. 2. MECM

3 A New RC Delay Metric Under Saturated Ramp Input

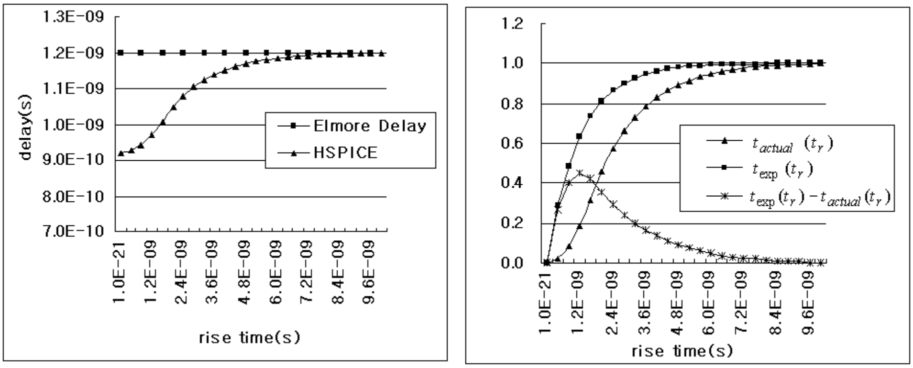
Figure 3(a) shows the relationship between rise time(t_r) and 50% propagation delay time(t_d) at an intermediate node of a circuit (node 7 of Figure 5 circuit) used for experimentation in Section 4. As expected as in previous work [11], t_d converges to T_{ED} as t_r increases.

As a result of experimentation with various RC-class interconnect circuits, it is evident that curves of almost all the HSPICE result graphs are similar to that shown in Figure 3(a), although the initial value(t_d at $t_r = 0$), final value(T_{ED}) and time-point of convergence are different. Defining that delay time at $t_r = 0$ is t_{step} and delay time under arbitrary ramp input is t_{ramp}, the main idea of this section is that we derive the expression, $t_{ramp} = f(t_r, t_{step}, T_{ED})$, resulting in a simple closed-form.

A delay metric for the computation of t_{step} can be selected by the trade-off between accuracy and simulation time. For an analysis in which accuracy is important, a complex yet accurate delay metric, e.g., h-gamma, can be used. For an analysis in which fast estimation time is required, a simple and fast delay metric, e.g., MECM, proposed in Section 2 can be used.

In this section, we assume that t_{step} has been computed by a suitable delay metric. Also, T_{ED} of an arbitrary node can be computed easily, using resistance and capacitance of interconnects [12]. We will derive a closed-form expression approximating the t_d curve in Figure 3(a) as a function of t_{step} and T_{ED}.

First of all, we normalize the graph of t_d curve to a new graph, called $t_{actual}(t_r)$, whose initial value is zero and final value is one, for the sake of easy derivation. Then, a simple expression of an exponential function as in Eq.(3), whose initial value is zero and final value is one, can be considered for t_d as a function of t_r.



(a) (b)

Fig. 3. (a)feature of graph for $t_d = f(t_r)$, (b)$t_{actual}(t_r)$,$t_{exp}(t_r)$,$t_{exp}(t_r) - t_{actual}(t_r)$

$$t_d = 1 - e^{\frac{-t_r}{\tau}} \tag{3}$$

Defining that Eq.(3) is $t_{exp}(t_r)$ and time constant, τ, is Elmore Delay, Eq.(4) can be derived.

$$t_{exp}(t_r) = 1 - e^{\frac{-t_r}{T_{ED}}} \tag{4}$$

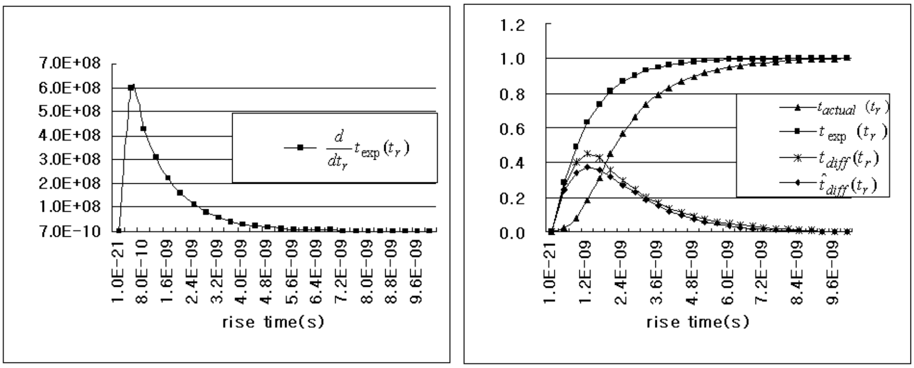
The graphs of $t_{actual}(t_r)$, $t_{exp}(t_r)$ and $t_{exp}(t_r) - t_{actual}(t_r)$ at node 7 of Figure 5 are shown in Figure 3(b). Defining that $t_{exp}(t_r) - t_{actual}(t_r)$ is $t_{diff}(t_r)$, the relation of the above graphs is represented as in Eq.(5).

$$t_{actual}(t_r) = t_{exp}(t_r) - t_{diff}(t_r) \tag{5}$$

Defining that the approximation of $t_{actual}(t_r)$ is $t_{ramp}(t_r)$ and the approximation of $t_{diff}(t_r)$ is $\hat{t}_{diff}(t_r)$, Eq.(5) can be represented as Eq.(6).

$$t_{ramp}(t_r) = t_{exp}(t_r) - \hat{t}_{diff}(t_r) \tag{6}$$

Now we can derive $t_{ramp}(t_r)$ by working on $\hat{t}_{diff}(t_r)$. We utilize a derivative of $t_{exp}(t_r)$ to obtain $\hat{t}_{diff}(t_r)$. Generally, when a derivative of an arbitrary function, $f(x)$ is $f'(x)$, a normalized function of $f'(x)$ is represented as $xf'(x)$. Hence, a derivative of $t_{exp}(t_r)$ can be normalized this way.



(a) (b)

Fig. 4. (a)$\frac{d}{dt_r}t_{exp}(t_r)$, (b)$t_{actual}(t_r)$,$t_{exp}(t_r)$,$t_{diff}(t_r)$,$\hat{t}_{diff}(t_r)$

The graph of $\frac{d}{dt_r}t_{exp}(t_r)$ is shown in Figure 4(a) and the normalized function of $\frac{d}{dt_r}t_{exp}(t_r)$ is represented as "$\hat{t}_{diff}(t_r)$" in Figure 4(b).

Using $\hat{t}_{diff}(t_r)$, Eq.(6) is modified as in Eq.(7).

$$\begin{aligned} t_{ramp}(t_r) &= t_{exp}(t_r) - \hat{t}_{diff}(t_r) \\ &= t_{exp}(t_r) - t_r \cdot \frac{d}{dt_r}t_{exp}(t_r)) \\ &= (1 - e^{\frac{-t_r}{T_{ED}}}) - (\frac{t_r}{T_{ED}}e^{\frac{-t_r}{T_{ED}}}) \\ &= 1 - (1 + \frac{t_r}{T_{ED}})e^{\frac{-t_r}{T_{ED}}} \end{aligned} \tag{7}$$

Since Eq.(7) is the normalized form, it represents the expression whose initial value is zero and final value is one. However, our final goal is to get the expression whose initial value is t_{step} and final value is T_{ED}. Therefore, the normalized graph should be rescaled to $(T_{ED} - t_{step})$ on y-axis and then be moved to t_{step} on y-axis. As a result, we can obtain a simple expression for t_{ramp} as in Eq.(8), which is the proposed FDM(Fast Delay Metric).

$$
\begin{aligned}
t_{ramp}(t_r) &= (1-(1+\frac{t_r}{T_{ED}})e^{\frac{-t_r}{T_{ED}}})(T_{ED}-t_{step})+t_{step} \\
&= T_{ED}-((1+\frac{t_r}{T_{ED}})e^{\frac{-t_r}{T_{ED}}})(T_{ED}-t_{step}) \qquad (8)
\end{aligned}
$$

4 Experimental Results

Delay times of interconnects vary significantly as the topology, resistance and capacitance of interconnects change. To keep the generality of test circuits, hence, we use the circuits experimented in the previous researches [2], [4], [7] as shown in Figure 5.

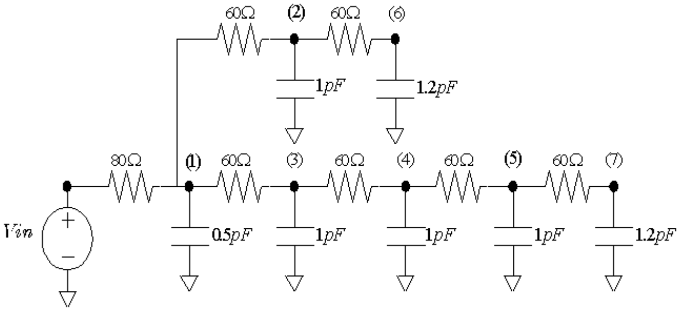


Fig. 5. An example 7-node RC circuit

4.1 Experiments for MECM

Table 1 shows the relative errors of delay times under step input at each node of Figure 5. The delay times using the previous delay metrics have been referred from [4], and (-) in the table implies the tendency of underestimating delay times.

The accuracy of MECM is superior to ECM and is similar to D2M which is relatively simple, yet accurate among the previous works using circuit moments.

Table 1. Relative errors(%) of each delay metric compared to HSPICE for the RC tree in Figure 5

	Delay Metric						
Node	MECM	ECM	D2M	Elmore	h-γ	DM1	DM2
1	46.3	69.1	51.9	180.4	-1.4	72.2	193.1
2	12.2	17.3	12.4	82.7	-5.1	17.5	57.2
3	10.2	-1.7	7.7	68.7	1.9	10.5	36
4	2.3	-12.2	-0.6	42.1	0	-0.6	-3.2
5	0.4	-12.7	-1.7	33.4	3.4	-0.7	-18.5
6	8.7	12.8	8.9	67	-4.7	10.5	30.4
7	0.1	-3.1	-1.5	30.5	-0.8	1.8	-24.9

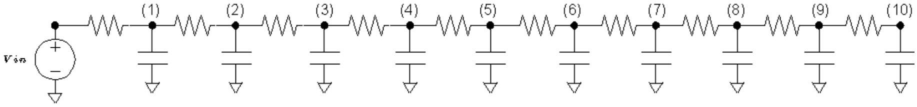


Fig. 6. Ten-segment RC ladder

Table 2. Relative errors(%) of each delay metric compared to HSPICE, averaged for 100 random ten-segment RC circuits

	Delay Metric					
Node	MECM	ECM	D2M	Elmore	DM1	DM2
1	292.6	314.7	273.4	1212.8	625	3385.3
2	117.9	132	115.2	377.1	180.9	553.8
3	53.1	64.3	54.3	183.9	74.9	198.1
4	30.1	32.9	24.6	104.9	31.5	80.4
5	14.1	17.6	10.3	68.4	12.4	30.3
6	8	9.9	3.3	49.5	3.7	4.3
7	5.6	6.4	0.6	39.8	1.1	10.4
8	4.7	5	0.1	35.4	1.4	18
9	4.3	4.5	0.1	33.2	2.9	22.2
10	4.1	4.3	0	32.3	4.1	24.3

Also, to verify the accuracy of MECM for the case of circuits having significant resistive shielding, we have experimented the method proposed by [4]. The test circuit is ten-segment RC ladder as shown in Figure 6, and each resistor and capacitor is randomly chosen between $1k\Omega \sim 20k\Omega$ and $1fF \sim 20fF$, respectively. We have generated 100 random circuits and averaged relative errors of delay times and then, the results compared with the previous delay metrics are shown in Table 2. The data for the previous delay metrics have been referred from [4]. As shown in Table 2, the accuracy of MECM is again similar to D2M.

4.2 Experiments for FDM

4.2.1 Case using MECM to compute step delay

Table 3 presents the relative errors of delay times computed using FDM at each node of Figure 5. Delay times under step input to be applied to FDM is computed by MECM, and the rise time of input signal is in the range of 0ns and 5ns. The more the experimented node is close to far-end nodes and the more the input signal rise time is lengthened, the more the accuracy of FDM(using MECM) increases. The error of FDM at near-end nodes is caused by the error of MECM. Estimation of delay times by combining FDM with MECM will show the best performance in timing analysis in which accurate estimation of far-end nodes is important and fast estimation time is required.

Table 3. Relative errors(%) of FDM using MECM compared to HSPICE for each node in Figure 5

t_r	Node						
	1	2	3	4	5	6	7
0ns	46.3	12.2	10.2	2.3	0.4	8.7	0.1
200ps	49.3	13.6	11.2	2.8	0.7	9.5	0.4
400ps	49.9	16	13.4	3.8	1.4	10.9	0.9
600ps	37.2	17.5	15.5	5.1	2.1	11.9	1.4
800ps	30.6	17.1	16.7	6.2	2.9	11.7	2
1ns	27.2	14.8	16	7	3.4	10.6	2.4
1.5ns	22.5	12.1	11.1	6.4	3.5	8.4	2.3
2ns	18.6	10.7	8.9	4.4	2.2	7.6	1.3
3ns	11.7	7.8	6.4	3.1	1.3	5.9	0.5
4ns	7	5.1	4.4	2.4	1.1	4.1	0.5
5ns	4.1	3.1	2.9	1.7	0.9	2.7	0.5

Table 4. Delay comparisons for each node in Figure 5 using RICE as the ideal delay metric(in ps)

rise time	250			500			1000			2000		
node	FDM	PERI	RICE	FDM	PERI	RICE	FDM	PERI	RICE	FDM	PERI	RICE
1	224	207	210	279	235	272	389	321	371	508	465	466
2	390	383	383	426	407	409	507	479	499	619	604	597
3	489	484	482	518	505	498	592	572	578	709	702	699
4	708	707	705	727	724	716	779	781	761	876	896	884
5	851	851	849	866	867	859	908	921	900	995	1030	1014
6	465	461	461	495	484	487	568	555	570	677	678	668
7	924	925	923	938	941	933	976	994	974	1058	1102	1086

4.2.2 Comparison of FDM and PERI

Table 4 shows a comparison between the FDM and experimental data presented in a recent work [7], for estimating delay times under ramp input. The PERI in Table 4 is the method proposed in [7] and utilizes RICE(4-pole) [13] to compute delay times under step input. Since the FDM is compared with the PERI, delay times under step input used in the FDM have also been provided from RICE algorithm.

As shown in Table 4, the accuracy of FDM is similar to that of PERI at all the range of rise times and all the nodes. This validates that the performance of FDM is close to PERI even though using FDM does not use circuit moments.

5 Conclusions

We have proposed an analytic technique, FDM, to estimate delay times under ramp input in a simple closed-form expression. The FDM is the technique extending delay metrics for step inputs to realistic non-step inputs easily. The FDM can estimate delay times under ramp input easily without using circuit moments, on the other hand, the PERI uses two circuit moments. To verify the validity of FDM, we have experimented with the same condition as in PERI and shown that FDM supports high accuracy, in spite of having the lower computational complexity than PERI.

In addition, to improve the disadvantages of previous delay metrics, we have proposed a new RC delay metric under step input, called MECM. We have compared the MECM with the previous delay metrics and validated that the accuracy of MECM is at least as same as that of the previous ones. Therefore, delay estimation by combining FDM with MECM will yield fast analysis time. Also, an estimation by combining FDM or MECM with the previous delay metrics may deal with various requirements (accuracy or estimation time or their compromise) in timing analysis flexibly, and we expect that the proposed delay metrics improve the performance of CAD tools requiring several millions of delay calculations for timing verification.

Acknowledgements

This work was supported by grant No. R01-2005-000-11215-0(2005) from the Basic Research Program of the Korea Science & Engineering Foundation.

References

1. A. B. Kahng, S. Muddu: An analytical delay model for RLC interconnects. IEEE Trans. Computer-Aided Design, vol. 16, pp.1507-1514, Dec. 1997
2. R. Kay, L. T. Pileggi: PRIMO: Probability interpretation of moments for delay calculation. in Proc. IEEE/ACM Design Automation Conference, June 1998, pp.463-468
3. T. Lin, E. Acar, L. T. Pileggi: h-gamma: An RC delay metric based on a gamma distribution approximation to the homogeneous response. in Proc. IEEE/ACM Int. Conf. Computer-Aided Design, Nov. 1998, pp.19-25
4. C. J. Alpert, A. Devgan, C. Kashyap: RC Delay Metrics for Performance Optimization. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol 20, pp.571-582, May 2001
5. N. Menezes, S. Pullela, F. Dartu, L. T. Pillage: RC Interconnect Synthesis - A Moment Fitting Approach. Proc. IEEE/ACM Intl. Conf. Computer-Aided Design, Nov. 1994, pp. 418-425
6. A. B. Kahng, S. Muddu: Analysis of RC Interconnections Under Ramp Input. UCLA CS Dept. TR-960013, April 1996
7. C. V. Kashyap, C. J. Alpert, Frank Liu, A. Devgan: Closed Form Expressions for Extending Step Delay and Slew Metrics to Ramp Inputs. ACM/SIGDA 2003 International Symposium on Physical Design, April. 2003
8. P. R. O'Brien, T. L. Savarino: Modeling the driving-point characteristic of resistive interconnect for accurate delay estimation. in Proc. IEEE/ACM Int. Conf. Computer-Aided Design, Nov. 1989, pp.512-515
9. Jessica Qian, Satyamurthy Pullela, Lawrence T. Pillage: Modeling the "Effective Capacitance" for the RC Interconnect of CMOS Gates. IEEE Trans. on Computer-Aided Design of Integrated Circuits and System, vol. 13, no. 12, Dec. 1994

10. A. B. Kahng, S. Muddu: Efficient Gate Delay Modeling for Large Interconnect Loads. IEEE Multi-Chip Module Conf., Feb. 1996
11. R. Gupta, Bogdan Tutuianu, Lawrence T. Pileggi: The Elmore Delay as a Bound for RC Trees with Generalized Input Signals. ACM/IEEE Design Automation Conference, June 1995, pp.364-369
12. J. Rubinstein, P. Penfield, Jr., M. A. Horowitz: Signal delay in RC tree networks. IEEE Trans. on Computer Aided Design, vol. 2, pp.202-211, 1983
13. C. L. Ratzlaff, N. Gopal, L. T. Pillage: RICE: Rapid interconnect circuit evaluator. in Proc. IEEE/ACM Design Automation Conf., June 1991, pp.555-560

Low Power Microprocessor Design for Embedded Systems*

Seong-Won Lee[1], Neungsoo Park[2], and Jean-Luc Gaudiot[3]

[1] Dept. of Computer Engineering,
Kwangwoon University, Seoul, Korea
swlee@kw.ac.kr
[2] Dept. of Computer Engineering,
Konkuk University, Seoul, Korea
neungsoo@konkuk.ac.kr
[3] Dept. of Electrical Engineering and Computer Science,
University of California, Irvine, California, USA
gaudiot@uci.edu

Abstract. Continuing advances in VLSI technology render a billion-transistor SOC device inevitable in the near future. However, along with this opportunity the excessive amount of power that billions of transistors will consume will be the most important challenge to the design of the future chips. Many techniques have been developed in order to reduce the power consumption of microprocessors. Unfortunately, this often comes at the expense of performance. In this paper, we describe a number of techniques which are currently used when designing low power, high performance microprocessors. These include fabrication process, circuit technology, and microprocessor architecture. Since most techniques result in complex tradeoffs, we will show how decisions regarding the selection of a low power design approach require careful consideration.

1 Introduction

Since the first appearance of microprocessors, many hardware techniques have been developed in order to improve their performance. Nowadays, emerging fabrication technologies even make a billion-transistor processor feasible. However, the power consumption generated by a billion transistors in a microprocessor will shortly become one of the main obstacles to the design of such large machines due to its high power consumption. The efforts at conserving power which had been mostly targeted at embedded systems in order to save battery power, increase mission time, *etc.* must now be geared towards reducing the wafer operating temperature to within reasonable limits [1]. Previous approaches to lower the power consumption of microprocessors largely concentrated on architecture-independent techniques such as low power fabrication technology (*e.g.,* Silicon

* This work was supported in part by the Ministry of Information and Communication, Korea, under the ITRC program supervised by the IITA and in part by the Research Grant of Kwangwoon University in 2005.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 622–630, 2006.

On Insulator [2]), scaling voltage and/or frequency according to the level of power consumption [3], and gating the clock signal to inoperative logic [4]. In truth, several of these techniques can be simultaneously used.

More recently, in addition to architecture-independent low power technologies, architectural innovations such as clustered architectures [5] have also been seriously studied. Indeed, it is well known that partitioning the microarchitecture into several clusters decreases power consumption by reducing the issue width of a superscalar architecture. Power management is another architectural approach for high performance microprocessors. Its role in high performance microprocessors remains to prevent damage to the chip due to the occasional heat generated. This is done by constraining the functionality of the microprocessor. Once a good power management scheme has been designed to take care of the excessive heat, designers could concentrate on improving performance.

In summary, power issues for microprocessors should be considered throughout the design process. It is the goal of this paper to describe, compare, and contrast the many power issues one encounters in microprocessor design at all the different stages. We begin by describing the various low power design layers in Section 2. Moving up one level, the efficiency of microprocessor architectures in terms of power consumption is discussed in Section 3. A design technique at the architectural level, the clustering technique, is introduced in Section 4. Another architectural level technique, power management, is discussed in Section 5. Software techniques are discussed in Section 6. Finally, we conclude and summarize the possible techniques of low power design of microprocessors in Section 7.

2 Design Layers for Low Power

Most work on managing power consumption in modern microprocessors has focused on reducing power consumption while keeping performance at acceptable levels. Designing a low power processor requires addressing all aspects of the design process from the selection of a low power technology to the architectural choice. However, it should be noted that, once the microprocessor has been fully optimized for speed, any attempt at reducing power consumption will inevitably cause a degradation in performance. Hence, low power method must be carefully selected and balanced against design for performance. This tradeoff will become more critical when, as we have seen, high power consumption will translate into unacceptable device temperatures. There are four layers of power-aware design methodologies. Those four design layers are mostly independent of each other, and can thus be often applied together. Let us now describe them in more detail and provide some examples.

- **Technology Solutions**
  Once the whole circuit has been designed, transition to finer design rules reduces the power consumption, *if we keep the same functionality.* However, it should be noted that *deep submicron technology* does not solve all problems:

for one thing, it encourages the addition of more functionality which correspondingly increases the total power consumption. Technology solutions include:

- *Low voltage technology* is now quite close to the threshold voltage which can physically turn on the transistors and is limited in its future applicability.
- *Silicon-on-insulator (SOI)* is one of the state-of-the-art CMOS technologies. The small junction capacitance of SOI reduces gate delay and power consumption. However, the floating body effect of SOI makes design difficult [2].

– **Layout and Circuit Techniques**
Low power circuitry such as low-power flip-flops can be used to lower power consumption. However, low power logic often has low performance. In order to avoid the low power logic's degradation of the performance of microprocessors, *dual voltage circuits* [6] (high voltage circuitry for the fast response that is necessary only for the critical signal path and low voltage circuitry for the non-critical path) can be used. Another technique is *clock gating* [7, 4] in which the clock is effectively not supplied to idle units. These techniques have disadvantages due to difficulties in the design of Power and clock distribution lines.

– **Architectural Decisions**
Different microprocessor architectures such as in-order execution and out-of-order execution have varying efficiency with regards to power consumption. According to Gonzalez *et al.* [8], *pipelining* increases the efficiency by a large amount, while out-of-order execution increases the efficiency by only a relatively small amount. The targets of a *restructuring of the microarchitecture* include the number of functional units, the issue bandwidth, the cache size, *etc.* Architectural variations such as *clustered microarchitecture superscalar* [5] are also possible. The *power management scheme* is also important because optimization for relatively uncommon situations such as peak power is not efficient for other cases.

– **System Softwares**
The whole design task of many modern devices is completed with its operating software. In addition to those hardware layers, low power software solutions for compiler and operating systems can also be considered [9]. Knowing the power consumption imposed by each instruction in the instruction set helps the compiler increase performance while satisfying the power constraints [10]. For example, reducing the total number of instructions while maintaining instantaneous power consumption below a certain level reduces the total energy consumed. Job scheduling based on power consumption of running programs also helps keep the power consumption of the system low [11].

3 Power-Efficiency of Microprocessor Architectures

In general, there is a strong relationship between power consumption and performance. The large number of transistors in modern microprocessors is mostly used to exploit Instruction Level Parallelism in superscalar architectures. However, due to the inherent limits of parallelism among instructions, single thread models cannot efficiently utilize resources in superscalar processors.

It is easiest for a compiler to effectively manage the power consumption in Very Long Instruction Word (VLIW) architectures [10]. Code Morphing techniques [12] show that well-optimized codes efficiently utilize resources and that consequently a typical VLIW processor would consume less power. However, issues regarding the design of a compiler sophisticated enough to efficiently utilize all resources and the compatibility of Instruction Set Architectures remain to be addressed.

On the other hand, we can improve the performance of microprocessors by exploiting Thread Level Parallelism (TLP) [13]. A single-Chip Multiprocessor (CMP) has several small microprocessors on a single silicon die [14]. Running multiple threads in parallel on a CMP returns a significant performance gain because of the exploitation of TLP in addition to ILP. The small instruction issue width of a CMP also contributes to a lower power consumption and to a possible increase of the clock frequency [15]. Furthermore, one can turn off standby modules with on-chip power switches. However, CMP does not have enough resources to exploit Instruction Level Parallelism within a thread and has low performance if only few threads are present.

Yet another approach, multithreading, can exploit Instruction-Level Parallelism as well as Thread-Level Parallelism because of its ability to share resources [16, 13]. Simultaneous MultiThreading (SMT) has been proposed as an architectural technique whose goal it is to efficiently utilize the resources of a superscalar machine without introducing excessive additional control overhead [17]. From an architectural point of view, the SMT architecture is one of the most efficient architectures in terms of resource utilization. Unlike CMP however, a Simultaneous MultiThreading processor does not have standby modules with any running threads because most modules are shared. However, the high resource utilization of Simultaneous MultiThreading should lead to a better performance per unit power consumption [18].

4 Microarchitecture Clustering Techniques

One of the most significant factors which affect power consumption in superscalar architectures is the instruction issue width. Increasing the issue bandwidth increases not only the complexity of the instruction queue (which is also known as the issue window) but also the number of read/write ports in register files, the number of functional units, and the result bus width. In order to reduce the effect of a large instruction issue width, microarchitecture clustering techniques are used for microprocessor architectures [5, 19]. By appropriately clustering the microarchitecture, the power consumption decreases dramatically and the performance can be sustained if instruction dispatches are carefully scheduled in order to minimize the dependencies between clusters (*i.e.*, intercluster dependency).

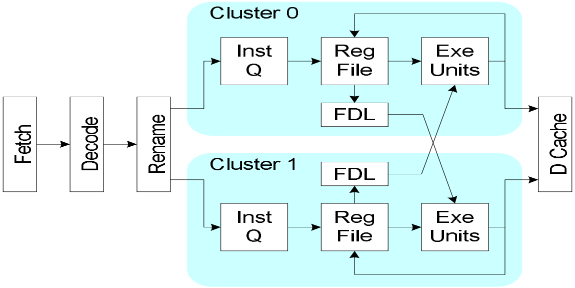


Fig. 1. Generalized Multiple Cluster Superscalar Architectures

With clustering techniques, the instruction issue width is divided into several small clusters. Figure 1 represents a generalized two-cluster superscalar architecture. Each cluster has its own issue window, register file, functional units, and result bus. Since superscalar architectures execute only a single thread at a time, there are heavy data and control dependencies between instructions in the two instruction queues. Forwarding logic (FDL in Figure 1) resolves those dependencies by transferring data from one cluster to the other.

The Alpha 21264 microprocessor [20] has such multiple clusters based on functional types of instructions, an integer type and a floating-point type. In addition to the conventional floating-point cluster, the integer execution units of the Alpha 21264 are classified into two identical clusters. Each integer cluster has its own full size register file synchronized with the register file in the other cluster. The synchronized full size register files resolve intercluster dependencies. The disadvantage of the clustered superscalar architecture is the performance degradation it incurs due to the underutilization of resources.

In order to avoid the underutilization of clusters due to the ratio of instruction types (the ratio of the number of integer over the number of floating-point instructions, in this case), some microarchitecture clustering techniques evenly partition the hardware resources of a microprocessor [5]. Since each cluster has both integer and floating-point functional units, instructions can be distributed to any cluster. However, fair instruction distribution and resolution of intercluster dependencies become crucial issues. Those two problems not only increase the complexity of the dispatch unit and forwarding logic but also demand heavy compiler support [21].

5 Power Management

As we have seen, as microprocessor fabrication reaches the deep submicron level, the total power consumption of a given architecture decreases *for a given functionality.* However, enlarging range of mobile applications introduce more strict power constraints into embedded systems. Therefore, even more dynamic power management schemes must be introduced in order to keep the heat produced by the microprocessor to within the desired operational range. These techniques consist of power or thermal detection and performance throttling. The dynamic power management techniques, include:

- **Clock Gating**
  In modern microprocessors, the clock distribution logic consumes a large portion of the total power consumption. Turning off the clock to the idle part of the processor can save significant amounts of power [6]. Because of their huge size, modern microprocessors requires a very complex clock distribution. In turn, this means that applying clock gating to all finer details in the circuitry of a microprocessor is often quite hard.
- **Global Voltage and Frequency Scaling**
  By checking the type of power supply, the supply voltage and the system clock frequency are controlled. Intel SpeedSetp [22] switches the voltage and the clock frequency between normal mode and low power mode. For example, if the battery power source is activated, the processor turns to the low voltage and low frequency mode. However, time critical applications may suffer from the lower performance.
- **Dynamic Voltage and Frequency Scaling**
  LongRun technology [12] by Transmeta changes the supply voltage and the clock frequency on the fly. A monitoring program picks an appropriate voltage and a clock frequency needed to run the application according to the performance demands. The dynamic voltage and frequency scaling needs to be extremely carefully designed because the dynamic clock control is complex with on-chip PLL and modules that operate with different clock rates [3].
- **Instruction Cache Throttling**
  While adopting the voltage and frequency scaling increases design complexity and constraints, PowerPC offers an alternative in the form of an architectural power management scheme [23]. In the PowerPC, instruction forwarding rates from the instruction cache to the instruction buffer can be controlled by setting the rate control register. Therefore, the idle time of the CPU increases and the overall temperature decreases.

Table 1. Summary of power management techniques

Power Management Scheme	Features
Clock Gating	Implementation is possible without architectural change. Fine calibration of clock distribution network is required.
Global Vtg/Freq Scaling	Design constraints due to V/F scaling is relatively loose. Response time is slow in the low power mode.
Dynamic Vtg/Freq Scaling	Effective power control with Fast response time. Timing constraints of each module is complicated because of fast V/F change.
On-chip Power Switches	Design and control of power management unit is simple. It is hard to turn off sub-modules individually.
I-Cache Throttling	Changing fetch policy in architectural level is fairly easy. Serious performance degradation if the fetch is a bottleneck.

According to Brennan *et al.* [6], the most effective way to save power is by changing the logic or architecture early, at the high-level design stage. Table 1 summarizes the power saving techniques we have described. It shows that most dynamic power management techniques sacrifice performance to save power. In order to preserve performance while saving power, one needs to analyze the pattern of power consumption in detail and then reduce the number of idle units.

6 Software Low Power Techniques

Nowadays, the performance of the processor and the size of memory have been drastically increased. These increments are usually accompanied by increasing the power consumption. Especially, the memory is one of significant factors increasing the power consumption. Most memory operations have relatively large power cost, both within processor and in the memory systems. Thus, the power management of the memory is one of the critical issues in the modern embedded system, since the portion of the memory system in the whole power consumption is drastically increased.

To reduce the power consumption in memory system, the number of memory accesses should be reduced. In order to achieve this goal memory access patterns are analyzed and classified to rearrange the sequence of memory access [24]. The page allocation is also considered. A cooperative hardware/software approach was proposed by Lebeck *et al.* [25, 26]. The proposed works explore the interaction between the page allocation in the view of software and the dynamic hardware control policies in the view of hardware. To improve the energy efficiency, the data access pattern is explored to allocate the page in the DRAM cooperating with the dynamic hardware power management policies.

Another software low power approach is resource scheduling technique. Since embedded systems must operates in various conditions of power restriction, the embedded real-time operating system (RTOS) should manage its resource adaptively. Tasks for the devices in the system can be rescheduled on the fly according to power conditions [27]. On the other hand, Pillai *et al.* [28] proposed a real-time scheduler for the RTOS that can regulate hardware power management techniques, especially dynamic voltage scaling.

7 Conclusion

High performance microprocessors have traditionally been developed for performance without much consideration to power consumption. The well known increase in the gate count of modern microprocessors correspondingly increases the amount of heat produced, so much that conventional cooling systems cannot effectively extract it. Hence, in the near future, the main design constraint in high performance microprocessor design has to become heat dissipation or its other form - power consumption.

Conventional low power technologies such as low power fabrication technology and clock gating techniques reduce the power consumption which is independent of the architecture of the microprocessor. In addition to architecture-independent technologies, architectural restructuring such as clustered microarchitecture techniques and its effectiveness on low power design are also serious contenders. In addition, As the power consumption of systems increases, system level techniques such as power management are emerging as the most important issue for embedded systems due to their adaptivity to various environments.

References

1. Dobberpuhl, D.: The Design of A High Performance Low Power Microprocessor. In: Proc. the 1996 Int'l Symp. Low-Power Electronics and Design. (1996) 11–16
2. Chuang, C., Lu, P., Anderson, C.: SOI for Digital CMOS VLSI: Design considerations and Advances. Proc. the IEEE **86** (1998) 689–720
3. Burd, T., Pering, T., Stratakos, A., Brodersen, R.: A Dynamic Voltage Scaled Microprocessor System. IEEE Journal of Solid-State Circuits (2000) 1571–1580
4. Gowan, M., Biro, L., Jackson, D.: Power Considerations in the Design of the Alpha 21264 Microprocessor. In: Proc. the 35th Design Automation Conference. (1998) 726–731
5. Zyuban, V., Kogge, P.: Inherently Lower-Power High-Performance Superscalar Architectures. IEEE Transactions on Computers **50** (2001) 268–285
6. Brennan, J., Dean, A., Kenyon, S., Ventrone, S.: Low Power Methodology and Design Techniques for Processor Design. In: Proc. the 1998 Int'l Symp. Low-Power Electronics and Design. (1998) 268–273
7. Bailey, D., Benschneider, B.: Clocking Design and Analysis for a 600-MHz Alpha Microprocessor. IEEE Journal of Solid-State Circuits **33** (1998)
8. Gonzalez, R., Horowitz, M.: Energy Dissipation In General Purpose Microprocessors. IEEE Journal of Solid-State Circuits **21** (1996) 1277–1284
9. Valluri, M., John, L.: Is Compiling for Performance == Compiling for Power? In: Proc. the 5th Annual Workshop on Interaction between Compilers and Computer Architectures (INTERACT-5). (2001)
10. Hwang, T., Lee, C., Lee, J., Tsai, S.: Compiler Optimization on Instruction Scheduling for Low Power. In: Int'l Symp. System Synthesis. (2000)
11. Chou, P., Liu, J., Li, D., Bagherzadeh, N.: IMPACCT: Methodology and Tools for Power-Aware Embedded Systems. In: Design Automation for Embedded Systems. (2002) 205–232
12. Klaiber, A.: The Technology Behind Crusoe Processors. (Transmeta Corporation)
13. Lo, J., Eggers, S., Emer, J., Levy, H., Stamm, R., Tullsen, D.: Converting Thread-Level Parallelism to Instruction-Level Parallelism via Simultaneous Multithreading. ACM Transactions on Computer Systems (1997) 322–354
14. Hammond, L., Nayfeh, B., Olukotun, K.: A Single-Chip Multiprocessor. IEEE Computer Special Issue on Billion-Transistor Processors **30** (1997) 79–85
15. Palacharla, S., Jouppi, N., Smith, J.: Complexity-Effective Superscalar Processors. In: Proc. the 24th Annual Int'l Symp. Computer Architecture. (1997) 206–218
16. Hirata, H., Kimura, K., Nagamine, S., Mochizuki, Y., Nishimura, A., Nakase, Y., Nishizawa, T.: An elementary processor architecture with simultaneous instruction issuing from multiple threads. In: Proc. the 19th Annual Int'l Symp. Computer Architecture. (1992)

17. Tullsen, D., Eggers, S., Emer, J., Levy, H., Lo, J., Stamm, R.: Exploiting Choice: Instruction Fetch and Issue on an Implementable Simultaneous Multithreading Processor. In: Proc. the 23rd Annual Int'l Symp. Computer Architecture. (1996) 191–202
18. Seng, J., Tullsen, D., Cai, G.: Power-Sensitive Multithreaded Architecture. In: Proc. the 2000 Int'l Conf. Computer Design. (2000) 119–206
19. Lee, S., Gaudiot, J.L.: Clustered Microarchitecture Simultaneous Multithreading. In Kosch, H., Böszörményi, L., Hellwagner, H., eds.: Euro-Par 2003. Parallel Processing, 9th Int'l Euro-Par Conference. Volume 2790 of Lecture Notes in Computer Science., Springer (2003) 576–585
20. Kessler, R.: The Alpha 21264 Microprocessor. IEEE Micro **19** (1999) 24–36
21. Farkas, K., Chow, P., Jouppi, N., Vranesic, Z.: The Multicluster Architecture: Reducing Cycle Time Through Partitioning. In: Proc. the 30th Annual Int'l Symp. Microarchitecture. (1997) 149–159
22. Intel Corporation: (Mobile Intel Pentium III Processor Datasheet)
23. Sanchez, H., Kuttanna, B., Olson, T., Alexander, M., Gerosa, G., Philip, R., Alvarez, J.: Thermal Management System for High Performance PowerPC Microprocessors. In: Proc. the 42nd IEEE Int'l Computer Conference, Motorola, Inc. and Apple Computer Corporation,USA (1997) 325–330
24. Kandemir, M., Sezer, U., Delaluz, V.: Improving Memory Energy Using Access Pattern Classification. In: Proc. Int'l Conf. Computer Aided Design. (2001) 201–206
25. Lebeck, A.R., Fan, X., Zeng, H., Ellis, C.: Power Aware Page Allocation. In: Proc. the 9th Int'l Conf. Architectural Support for Programming Languages and Operating Systems. (2000)
26. Fan, X., Ellis, C.S., Lebeck, A.R.: Memory Controller Pollicies for DRAM Power Management. (2001)
27. Lu, Y., Benini, L., Micheli, G.D.: Low-Power Task Scheduling for Multiple Devices. In: Proc. 8th Int'l Workshop on Hardware/Software Codesign. (2000) 39–43
28. Pillai, P., Shin, K.: Real-Time Dynamic Voltage Scaling for Low-Power Embedded Operating Systems. In: Proc. 18th Symp. Operating Systems Principles. (2001)

History Length Adjustable *gshare* Predictor for High-Performance Embedded Processor

Jong Wook Kwak[1], Seong Tae Jhang[2], and Chu Shik Jhon[1]

[1] Department of Electrical Engineering and Computer Science,
Seoul National University, Shilim-dong, Kwanak-gu, Seoul, Korea
{leoniss, csjhon}@panda.snu.ac.kr
[2] Department of Computer Science, The University of Suwon,
Suwon, Gyeonggi-do, Korea
stjhang@suwon.ac.kr

Abstract. As modern microprocessros and embedded processors employ deeper pipelines and issue multiple instructions per cycle, accurate branch predictors become an essential part of processor architectures. In this paper, we introduce a history length adjustable *gshare* predictor for the high-performance embedded processors and show its low-level implementation. Compared to the previous *gshare* predictor, history length adjustable *gshare* predictor selectively utilizes the branch history, resulting in substantial improvement in branch prediction accuracy.

Keywords: Branch Prediction, Branch History, History Length Adjustment, *gshare* Predictor.

1 Introduction

Modern microprocessors employ deeper pipelines and issue multiple instructions per cycle. Under these conditions, miss-predicted branch instructions require substantial amount of execution time to correct the execution path, resulting in considerable decrease of IPC. In the field of embedded processors, such technology trends are expected to be realized in a near future as well[1][2].

Until now, many dynamic branch predictors have been proposed, with their distinctive unique features. However, the complex hybrid branch predictors can not directly be implemented in embedded processors, due to the strict hardware resource constraints in its environments. Instead, most branch predictors in embedded processors still have used the small-scale *bimodal* style predictors[2]. However, prediction accuracy of branch predictors which utilize the *correlation* features, such as *GAg*, *gshare*, is expected to increase, as shown in many previous researches[3][4]. Compared to the complex hybrid branch predictors, *gshare* predictor has a relatively simple structure and its hardware requirement is quite modest. Further, although *gshare* predictors require more hardware budget than *bimodal* predictors, we think that the requirement will be moderated in a near future, and *gshare* predictor will become one of common choices in high-performance embedded processors.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 631–638, 2006.

In this paper, to additionally increase the branch prediction accuracy of *gshare* predictor, we propose history length adjustable *gshare* predictor and show its low-level implementation. The rest of this paper is organized as follows. Section 2 describes the related works about this paper. Section 3 proposes history length adjustable *gshare* predictors and show its low-level implementation. In section 4, we simulate our proposal and discuss the simulation results. Finally, section 5 concludes this paper.

2 Related Works

Originally, the branch address (PC) was firstly proposed to index the Pattern History Table (PHT). This style of predictors is often called the *bimodal* predictor and it is still widely used in current microarchitectures and embedded processors, because the bimodal execution of branch instruction is one of main characteristics of most branch instructions. However, since the introduction of two-level adaptive branch predictor, Global Branch History (GBH) has been a major input vector in branch prediction, together with the address of branch instruction[3]. Some branches, especially conditional branches, are strongly correlated to the outcomes of a few past branch instructions. In case of these branches, the GBH can efficiently utilize the correlation features of branch instruction, resulting in prediction accuracy enhancement. However, one of the main problems which cause the degradation of the prediction accuracy is PHT aliasing. It causes the prediction results of multiple branch instructions map into the same entry in the PHT. The first scheme to address the aliasing problem was the *gshare* predictor[4]. The idea of this predictor is the *exclusive-or* function between two input vectors (PC and GBH), and this function makes more even use of the PHT entries.

Besides, the length of GBH is statically fixed for all branch instructions in *gshare* predictors, and the history length is usually selected in accordance with the size of PHT. However, many previous works showed that different branch instructions require different length histories to achieve high prediction accuracies [5][6][7]. J. Kwak proposed the history length adjustment algorithm and the required hardware module[8]. The proposed solution tracks data dependencies of branch instructions and identifies strongly correlated branches(called *key branch*) in branch history, based on *operand register* in branch instruction. By identifying the key branch, it selectively uses the information of key branch in GBH, resulting in different history length for each branch instruction. In this paper, we make use of this idea, and we propose history length adjustable *gshare* predictor and show its low-level implementation, which provides the history length adjustment capability.

3 History Length Adjustment in *gshare* Predictor

In this section, we propose history length adjustable *gshare* predictors and show its low-level implementation. At first, Fig. 1 shows the original *gshare* predictor. The original *gshare* predictor utilizes the Program Counter (PC) and the Global Branch History (GBH) for each branch instruction. And then, the PHT is indexed by the *exclusive-or* function between these two input vectors. McFarling[4], who proposed the predictor, firstly observed that the *exclusive-or*ing the PC with GBH has more information than either component alone. That is, *exclusive-or*ing two input vectors produces more unique pattern to index the PHT.

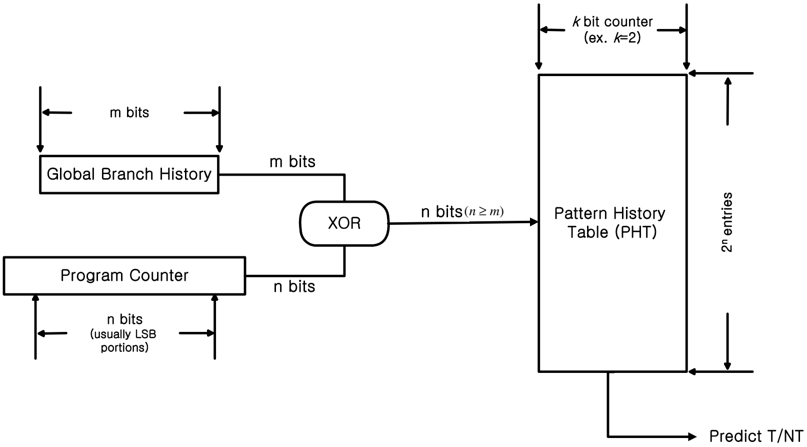


Fig. 1. *gshare* Predictor

Compared to Fig. 1, Fig. 2 shows the low-level implementation of *gshare* predictor. In Fig. 2, we assume that the GBH and the PC used are 8 bits and 10 bits, respectively. The GBH can be implemented by 8 bit shift register and each GBH field is *bit-wise exclusive-or*ed with the PC field.

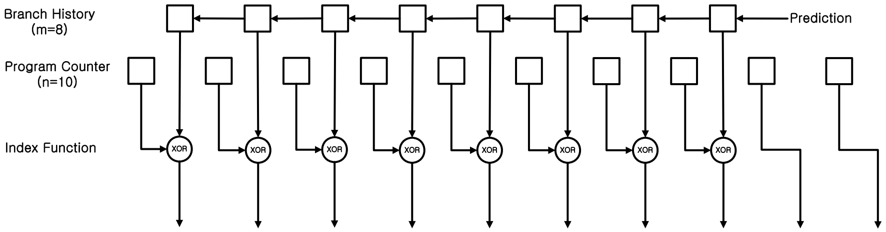


Fig. 2. Low-Level Implementation of *gshare* Predictor

The formal definition of the proposed predictor, the history length adjustable *gshare* predictor, is shown in Fig. 3. As shown in Fig. 3, the main operation of history length adjustable *gshare* predictor is composed of six steps. When history length adjustable *gshare* predictor predicts the branch instruction B_i in prediction time c, the address of $B_i(Address_i)$ is obtained in step 1. In step 2, the last

$$Step\,1:\ Address_i = A_{i,n}\,A_{i,n-1}\ldots A_{i,2}\,A_{i,1}$$

$$Step\,2:\ GHR_i = H_{i,c-m}\,H_{i,c-m+1}\ldots H_{i,c-2}\,H_{i,c-1}$$

$$Step\,3:\ H_c = \phi(GHR_i, Length\_Indicator_{\ i})$$

$$Step\,4:\ Q_c = \omega(H_c,\ Address_i),\ State_c = PHT_{Q_c}$$

$$Step\,5:\ Z_c = \lambda(\,State_C)$$

$$Step\,6:\ State_{c+1} = \delta(\,State_C, H_{i,c}\,)$$

Fig. 3. History Length Adjustable *gshare* Predictor

m outcomes of GBH are extracted and fed into $GHR(GlobalHistoryRegister)_i$. Then, history length adjusting function ϕ produces the adjusted history vector H_c using GHR_i and $Length\_Indicator_i$ in step 3, PHT index function ω selects the entry Q_c on PHT from H_c and $Address_i$ in step 4, and the prediction decision function λ predicts the prediction results Z_c for branch instruction B_i, using the counter value $State_c$ on PHT entry Q_c in step 5, sequentially. Finally, next state resolution function δ decides the next state ($State_{c+1}$) of Q_c entry on PHT with $State_c$ and $H_{i,c}$ in step 6.

Fig. 4 shows an implementation for the proposed predictor. The only difference from the original *gshare* predictor is the *and* function before the *xor* function. In Fig. 4, before executing *xor* function between GBH and PC, original GBH is fitted by Length_Indicator through the *and* function. At this time, we use the modified value of Length_Indicator by the Global History Enabler (GHE), then the fitted history is *bit-wise exclusive-or*ed with PC. In this manner, each field of GBH is selectively provided to index the PHT, by the combination of *and* function and *xor* function.

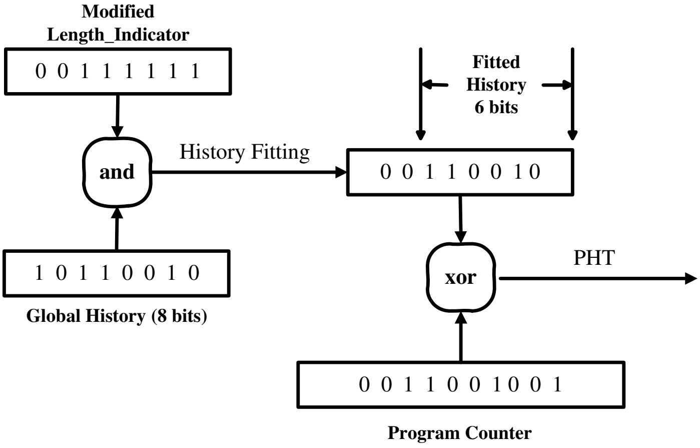


Fig. 4. History Length Adjustable *gshare* Predictor

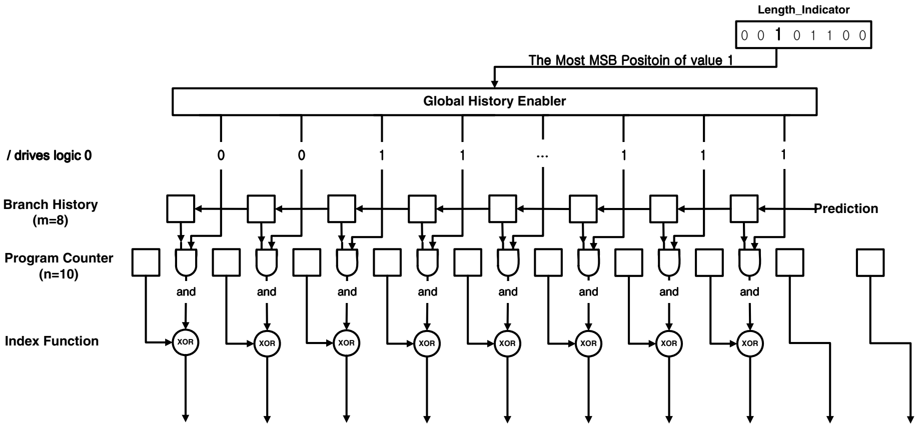


Fig. 5. Alternative Implementation for History Length Adjustable *gshare* Predictor

On the other hand, Fig. 5 shows low-level implementation of the history length adjustable *gshare* predictors. The additional hardware costs, compared to Fig. 2, are Global History Enabler (GHE) and *and* function. The main role of GHE is the modification of Length_Indicator value, as mentioned in above, to selectively provide the branch history. In GHE, the original bit vector of Length_Indicator is modified as follows; set each bit field as logic value 1 from the most MSB field of bit 1 to LSB field (ex., from 00***1***01100 to 00***1***11111). Then, *and* function selectively provides each GBH field.

4 Performance Evaluation

In this section, we evaluate the performance of our proposal. We use an event-driven simulator, *SimpleScalar*[9], for our simulation. As benchmark programs, we use *SPEC 2000* application suits[10]. Table 1 shows the overall system parameters and simulation environments.

At first, Fig. 6 and Fig.7 shows the miss-prediction rate(%) and IPC in 1K, and 2K PHT, respectively. In our simulations, *Length Adjustment* is the proposed solution in this paper. In addition, *Profiled Best* means the results of the best miss-prediction rate through the prior-profiling for each application, and *Fixed Length* means the results of the previous original *gshare* policy. As shown in Fig. 6 and Fig. 7, the proposed solution outperforms the *Fixed Length* policy for all applications, up to 5.6%. Further, the *Length Adjustment* policy, sometimes, even outperforms the *Profiled Best* policy, in case of *parser*, *eon*, *bzip2*.

Fig. 8 shows the average miss-prediction rate for all applications, and the results are shown according to the PHT size. In this results, we can more easily observe the performance of *Length Adjustment* policy and compare the performance of proposed solution with the *Profiled Best*. *Profiled Best* is generally considered as the optimal solution in many works. However, our policy even

Table 1. Simulation Parameters

System Parameter	Value
Fetch Queue	4 entries
Fetch and Decode Width	4 insructions
ROB entries	16 entries
LSQ entries	8 entires
Functional Units(Int and FP)	4 ALUs, 1Mult/Div
I-TLB	$64(16 \times 4ways)$ entries, 4K pages, 30 cycle miss
D-TLB	$128(32 \times 4ways)$ entries, 4K pages, 30 cycle miss
Predictor Style	bimodal, 2-level, gshare
BTB entries	$2048(512 \times 4ways)$ entries
RAS entries	8 entries
Extra Miss Penalty	3 cycles
L1 I-Cache	16KB, direct-map, 32B line, 1 cycle
L1 D-Cache	16KB, 4-ways, 32B line, 1 cycle
L2 Cache(Unified)	256KB, 4-ways, 64B line, 6 cycles
Memory Latency	first chunk=18 cycles, inter chunk=2 cycles

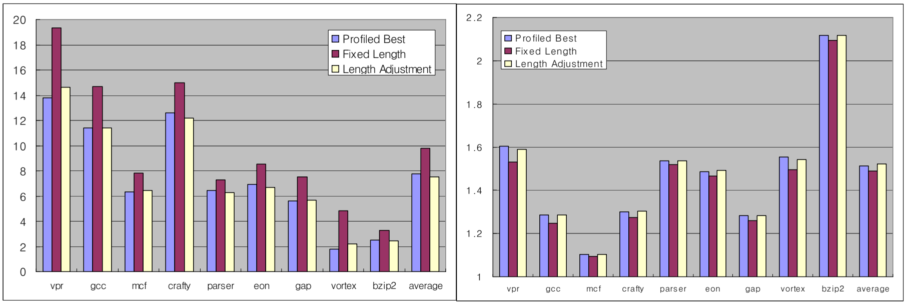


Fig. 6. Miss-Predictin Rate and IPC, 1K PHT

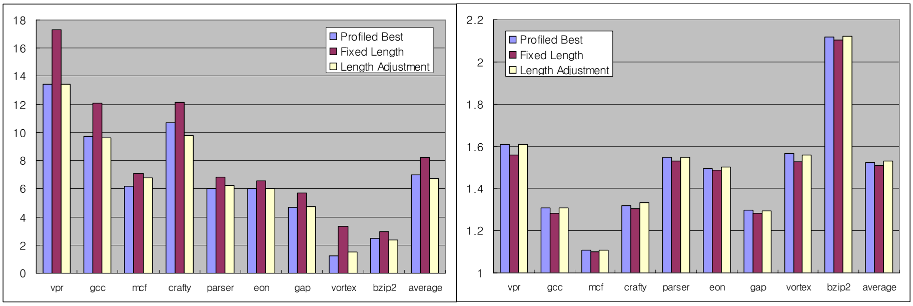


Fig. 7. Miss-Predictin Rate and IPC, 2K PHT

outperforms the optimal result. This result is mainly due to the following features of the proposed solution, such as *fully-dynamic* adjustment in program execution time as well as *per-branch* history length adjustment, regardless of application characteristics and system environments, to selectively provide the strongly correlated branch history.

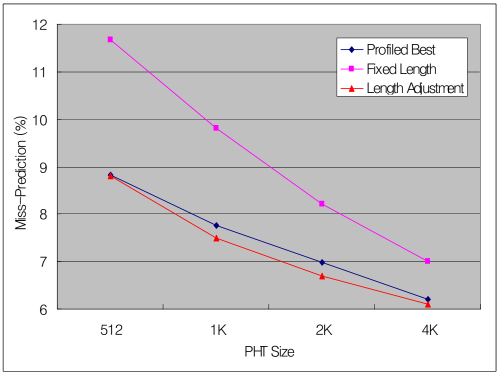


Fig. 8. Average Miss-Prediction Rate for Each PHT Size

5 Conclusion

The accurate branch predictors have become one of essential parts of modern microarchitectures and embedded processors. In this paper, we presented History Length Adjustable *gshare* predictors, which can dynamically change the history length used, and we showed its low-level implementation. In simulation parts, the proposed solution provides substantial prediction accuracy enhancement, compared to the original *gshare* predictor. Further, history length adjustable *gshare* predictor, sometimes, provides better prediction accuracy than the results of prior-profilings, which are generally considered as an optimal solution.

References

1. Steve Furber, “ARM System-on-Chip Architecture”, Second Edition, Addison-Wesley, 2000
2. Intel XScale Core Developer’s Manual, January 2004
3. Yeh, T. Y. and Patt, Y. N., “Two-level adaptive branch prediction”, In Proceedings of the 24th ACM/IEEE International Symposium on Microarchitecture, 51-61, 1991
4. McFarling, S., “Combining branch predictors. Tech. Rep. TN-36m”, Digital Western Research Lab., June, 1993
5. J. Stark, M. Evers, and Y. N. Patt, “Variable length path branch prediction”, In Proc. 8th Int’l Conf. on Architectural Support for Programming Languages and Operating Systems, pp. 170-179, 1998.

6. M.-D. Tarlescu, K. B. Theobald, and G. R. Gao, “Elastic history buffer: A low-cost method to improve branch prediction accuracy”, In Proc. Int’l Conf. on Computer Design, pp. 82-87, 1997.
7. T.Juan, S. Sanjeevan, and J. J. Navarro, “Dynamic history length fitting: A third level of adaptivity for branch prediction”, In Proc. 25th Int’l Symp. on Computer Architecture, pp. 155-166, 1998.
8. Jong Wook Kwak, “Effective Input Vector Management to Improve Branch Prediction Accuracy”, Ph. D. Thesis, Department of EECS, Seoul National University, 2006.
9. D. Burger, T. M. Austin, and S. Bennett, ”Evaluating future micro-processors: the SimpleScalar tool set”, Tech. Report TR-1308, Univ. of Wisconsin-Madison Computer Sciences Dept., 1997
10. SPEC CPU2000 Benchmarks, http://www.specbench.org

Security Engineering Methodology Based on Problem Solving Theory

Sangkyun Kim[1] and Hong Joo Lee[2]

[1] Yonsei University, Seoul, Korea
saviour@yonsei.ac.kr
[2] Dankook University, Seoul, Korea
blue1024@dankook.ac.kr

Abstract. This paper answers the difficult problems that organizations face in business environments when they try to solve information security issues by suggesting the integrated methodology for security engineering. Contributions of this paper are summarized as following. The first is the provision of requirements of security engineering methodology based on the model of ill-structured problem solving. The second is the framework which integrates various methods and tools of security engineering. The third is a suggestion of the process model and components which support an entire lifecycle of security management.

1 Problem Solving and Security Engineering

Professionals are hired and retained in most contexts in order to solve problems [1]. Problems cluster into three kinds of problems such as puzzle problems, well-structured problems, and ill-structured problems. Puzzle problems include content-neutral puzzles, such as anagrams, the Tower of Hanoi problem [2], and the Nine Dots problems [3]. These are domain-independent and not tied either to school practice or to real-world practice [4]. Well-structured problems are well-defined in that all elements and processes needed to solve the problems are present. They also have single correct, convergent answers and preferred solution processes [5]. Ill-structured problems are the kinds of problems that are encountered in everyday practice, so they are typically emergent dilemmas. However, real-world problems are often complex and ill-structured and therefore require different skills for successful solutions [6, 7]. The security problems of organization occurred in real-world should be managed as an ill-structured problem solving because the characteristics of security problems are similar to the features of ill-structured problems. These are as follows: 1) Have vaguely defined or unclear goals [8]. 2) Possess multiple solutions, solution paths, or no solutions [9]. 3) Have no prototypic cases because case elements are differentially important in different contexts and because they interact [10, 11]. Jonassen proposed a seven-step process of ill-structured problem solving [4]. In this paper, Jonassen's seven-step model is used to design the framework of integrated methodology for security engineering. Table 1 summarizes

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 639 – 648, 2006.

the relationships between Jonassen's model and the requirements of the integrated methodology for security engineering.

Table 1. Relationships between Jonassen's model and the methodology for security engineering

Jonassen's model	Requirements of the methodology for security engineering
Step-1: Identify the problem space	R-1: Analyze the risks of valuable information assets to decide if problems really exist
Step- 2 : Identify and clarify the alternative perspectives	R- 2 : Evaluate the current status of information security systems to clarify diverse perspectives of security problems
Step- 3 : Generate the solutions	R- 3 : Make strategic plans of information security systems which are the solutions for security problems
Step- 4 : Assess validity of possible solutions	R- 4 : Justify the economic value of possible security controls based on the comparison between a cost of security controls and effects
Step- 5 : Monitor the problem space and solution options	R-5: Estimate if the selected controls could improve the value of information security systems
Step- 6 : Implement and monitor	R-6: Select and introduce the proper security controls
Step- 7 : Adapt the solution	R-7: Operate security systems to provide feedback for adjustment and adaptation of current systems

2 Previous Researches

Previous researches on the methodology for security engineering are as follows: maturity model for information security, evaluation or auditing indices for ISMS, and management guidelines. The SSE-CMM is a compilation of the best-known security engineering practices [12]. It suggests some goals of security engineering are to [13]: Gain understanding of the security risks; Establish a balanced set of security; Transform security needs into security guidance; Establish confidence or assurance in the correctness and effectiveness of security mechanisms; Determine that operational impacts due to residual security vulnerabilities; Integrate the efforts of all engineering disciplines and specialties. The NIST handbook provides the assistance in securing computer-based resources by explaining important concepts, cost considerations, and interrelationships of security controls. It illustrates the benefits of security controls, the major techniques or approaches for each control, and important related considerations [14]. It consists of three controls: management controls, operational controls, and technical controls. BS7799 is the most widely recognized security standard in the world. It consists of ten major sections, each covering a different topic or area [15]. BS7799 provides general guidance on the wide variety of topics. BS7799 does not provide enough information to support an in-depth organizational information security review or to support a certification program. SC27 WG1 of ISO/IEC JTC1 suggested ISO/IEC TR13335. It consists of 5 parts: Part 1- Concepts and models for IT Security, Part 2- Managing and planning IT Security, Part 3- Techniques for the management of IT Security, Part 4- Selection of safeguards, Part 5- Application of IT security services and mechanisms [16].

Table 2 summarizes the objectives of previous researches and the relationships between previous researches and this paper. The column of relationship describes how previous researches are referred in this paper.

Table 2. Summary of previous research: objective & relationship

Category	Objective	Research	Relationship
International standards	Auditing of ISMS	[15]	Evaluation factors of ISMS
General methodology	Life cycle of security activities	[14, 16, 17]	Process model (roadmap) of this methodology
	Domain and capability of security engineering	[13]	Maturity model of ISMS
Evaluation and analysis	Risk analysis	[18, 19]	Analysis model of information assets for security strategy planning
	Assessment of ISMS	[20, 21]	Evaluation factors of ISMS
	Evaluation of security products	[22, 23, 24]	Evaluation and selection process of security controls
	Classification, decision and comparison factors	[25, 26, 27, 28 , 29, 30, 31, 32, 33]	Evaluation and selection factors of security controls
Economic justification	Concept	[34, 35, 36, 37]	Framework of economic justification of security investment
	Empirical study	[38, 39]	Process of economic justification of security investment
	Cost factors of security systems	[40, 41, 42]	Cost factors of economic justification of security investment

3 Framework of Integrated Methodology for Security Engineering

The framework of the integrated methodology consists of the patterns and scenarios (level 1), road map (level 2), and components (level 3) [12, 43, 44, 45, 46, 47, 48]. The framework is illustrated in figure 1.

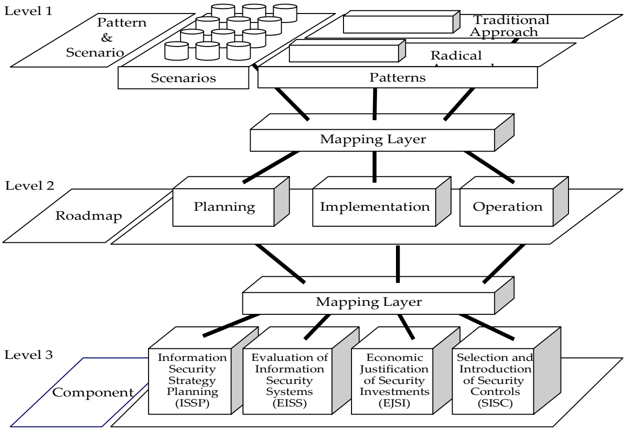

Level 1
Pattern & Scenario
Traditional Approach
Radical
Scenarios
Patterns
Mapping Layer
Level 2
Roadmap
Planning
Implementation
Operation
Mapping Layer
Level 3
Component
Information Security Strategy Planning (ISSP)
Evaluation of Information Security Systems (EISS)
Economic Justification of Security Investments (EJSI)
Selection and Introduction of Security Controls (SISC)

Fig. 1. Framework of the integrated methodology for security engineering

As described in section 1, the framework of this methodology is based on Jonassen's model of ill-structured problem solving. Level 1 defines organization-specific characteristics and generates a customized architecture of methodology. Level 2 provides a customized path from planning to operation according to the customized architecture generated in level 1. Level 3 provides functional components which support the path generated in level 2.

4 Level 1: Patterns and Scenarios

Scenarios define some special cases associated with the characteristics of organizations which need leading-edge systems for information security. The key milestones are specified by the scenarios which define special cases. Patterns define specific characteristics of an organization such as a scale of organization and scope of information security systems. The master road map is converted to the scenario-applied road map through a scenario database, and the scenario-applied road map is converted to the strategy-applied road map through the strategy specific patterns. Finally, the strategy-applied road map is converted to the character-applied road map through the character specific patterns, and it is the fully optimized road map for the organization.

5 Level 2: Roadmap

The road map is a set of classification and relation of tasks that support lifecycle of information security systems from planning to operation. The road map is also a set of objective functions that the organization must execute successfully to achieve competitiveness through information security systems, and it is purely related to business environments and implementation strategy. Therefore, the road map presented in this methodology should be customized into several paths by scenarios and patterns. The processes of the road map are illustrated in figure 2. This paper also suggests a certification process because there are practical regulations forced by international organizations and domestic agencies in many countries.

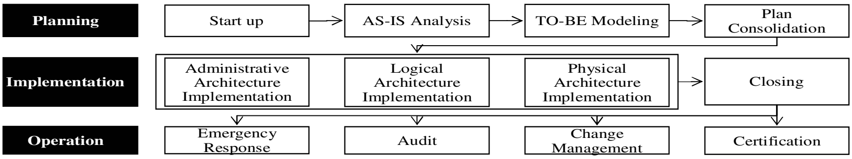


Fig. 2. Roadmap

6 Level 3: Component

This methodology suggests four kinds of component. These are: ISSP for information security strategic planning; EISS for evaluation on information security systems; EJSI for economic justifications on an investment on information security systems; SISC for selection and introduction of security controls. These components are designed by the requirements analysis of the integrated methodology for security engineering which is based on Jonassen's model for ill-structured problem solving.

6.1 Component: ISSP

Jeon suggested a relational framework between an AS-IS model and TO-BE model for information strategy planning methodology [49]. ISSP suggests a relational framework between an AS-IS model and TO-BE model based on Jeon's model and $(CIS)^2$ model [50]. The framework of ISSP is illustrated in figure 3.

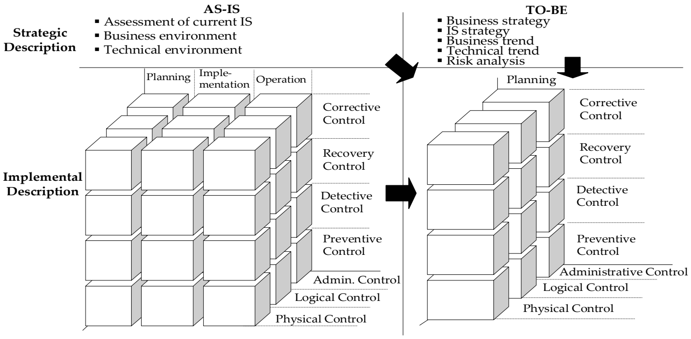


Fig. 3. Relational framework of an AS-IS and TO-BE model of information security strategy

The key characteristics of four sectors of this model are described in table 3.

Table 3. Key characteristics of transformation model

Dimension		Description
Strategic Description	AS-IS	AS-IS Strategic Description: An assessment of current information security systems introduces a brief and wide review of AS-IS status and their problems and requirements. A business environment analysis shows managerial risks. A technical environment analysis shows systematical risks.
	TO-BE	TO-BE Strategic Description: An analysis on a business strategy and IS strategy identify strategic requirements of information security systems. A business and technical trends analysis suggest operational requirements. A risk analysis prioritizes information assets and determines what should be secured first based on its value which is estimated by its importance on business capabilities, vulnerabilities, and potential threats.
Implemental Description	AS-IS	AS-IS Implemental Description: The existing information security strategy developed in last planning operation is analyzed to identify what is planned to be implemented. Implemented models are analyzed to validate if these were implemented as planned in strategy documents. An operation status is reviewed including a change management, emergency response, audit, and certification to assess operational risks.
	TO-BE	TO-BE Implemental Description: The implemental description of TO-BE model is a strategy planning. It only concerns on a modeling of administrative, logical, and physical controls and their implementation plan.

6.2 Component: EISS

The evaluation factors consist of two categories: procedures and supporting systems. The evaluation factors of procedures are derived from Leem's model: completeness, validity, consistency and feasibility [51]. Table 4 shows the evaluation criteria of procedures for information security planning. The evaluation factors of supporting systems are derived from Leem's model: organizational supports, investment, evaluation system and education system [51]. Table 5 shows the evaluation criteria of supporting systems for information security planning.

Table 4. Evaluation criteria of procedures for planning

	Completeness	Validity	Consistency	Feasibility
Business Strategy Analysis	Business mission, objectives, CSF	Precision of analysis	Alignment with TO-BE scope & goal	Accuracy of gathered information
Environment Analysis	Competitive environments and technical environments analysis	Concreteness of SWOT analysis; Precision of analysis	Consistency in environmental analysis	Accuracy of gathered information
TO-BE Scope & Goal	Scope; Goal; Objectives	Concreteness of objective and scope of planning; Concreteness of mission and CSF	Consistency with TO-BE modeling	Compliance check
AS-IS Model Assessment	Administrative model; Logical model; Physical model	Precision of analysis	Consistency in each component of (CIS)2 model	Accuracy of gathered information
Risk Analysis	Asset classification; Threat, vulnerability identification	Methodology definition; Precision of decision on priority	Alignment between asset and their threat	Clearance of asset, threat
TO-BE Modeling	Administrative model; Logical mode; Physical model; Review and evaluation of plans	Precision of analysis of improved models; Precision of review of plans	Consistency between modeling structure and risk analysis; Consistency among improved models; Consistency in evaluation results with modification of plan	Accuracy of evaluation criteria; Adequacy of evaluation methods; Confirmation by the user
Implementation Planning	Project plan; Approval; Education plan; Maintenance plan	Concreteness of IT project, education plan, and maintenance plan	Consistency in implemental road map of information security plan with IS or business strategy	Feasibility of time, cost, quality, human resource, communication, procurement, education and maintenance plan

Table 5. Evaluation criteria of supporting systems for planning

Dimension	Criterion
Organizational supports	Organizational position of the manager who is responsible for project; Extent of top managements' commitment; Extent of users' commitment
Investments in information security	Duration of Implementation; Extent of material/human resources allocated; Extent of investments in each solution
Evaluation of information security & plans	Quality of performance measure for information security & plans; Quality of quantitative measure for effectiveness; Quality of operation and maintenance plan
Education system	Quality of education program; Extent of investments on education; Education ratio of total employees on information security

6.3 Component: EJSI

The cost factors consist of nine groups with two perspectives of the lifecycle and control category. The control category is derived from previous researches on security controls [25, 26, 27, 28, 29]. The cost factors are provided in table 6. The benefit factors consist of the operational benefits and the strategic benefits. The operational or strategic benefits may be categorized into one of three types of expression factor: economic factor which is measured and evaluated with monetary terms, numerical factor which is measured and evaluated with number or volume, and qualitative factor. Operational benefits mean the enhanced efficiency of organization's operations. The strategic benefits mean enhanced competitive advantages. According to Porter's five competitive forces model, there are five threats such as the threat of new entrants, the power of suppliers, the threat of substitute products, and the rivalry among existing competitors [52]. The benefit factors are described in table 7.

Table 6. Cost factors of a security investment

	Administrative	Logical	Physical
Planning	Loss of working; Staffing(planning TFT); Consulting; Awareness/training/education	Computing; Comm. equipment; Analysis tool; System downtime	Space; Supporting utility; Tempest/shielding; Monitoring/alarming
Implementation	Loss of working; Public charge; Staffing (implementation TFT); Outsourcing (app. development, system build up); Awareness/training/education	OS; S/W; H/W; DB; Contents; Comm. equipment; System downtime	Space; Supporting utility; Tempest/shielding; Monitoring/alarming
Operation	Insurance; Public charge; Staffing; Certification; Awareness/training/education; Loss of productivity	Insurance; Maintenance; Repair/ replace/upgrade	Insurance; Maintenance; Repair/ replace/upgrade

Table 7. Benefit factors

Operational Factor	Measurement index	Measurement factor
Cost saving (Prevention of potential losses)	Revenue	Direct losses; Compensation payment; Billing losses; Investment losses
	Service cost saving	A/S cost saving
	Firm infrastructure saving	Equipment rental saving; Space rental cost saving; Shipping cost saving
	Human resource cost saving	Prevention of potential work losses; Temporary employment saving; Overtime cost saving
	Financial performance	Lost discounts
Added profitability	Increase of sales	New customer occurrence; Existing customer preservation
	Increase of profitability	Added premium value; Enhanced productivity
Enhanced decision making	Time reduction	Reduced decision making time; Reduced decision making step
	Enhanced quality	Enhanced problem cognition; Enhanced correctness
Enhanced business function	Enhanced flexibility	
	Enhanced credibility	
Strategic Factor	Measurement Index	Measurement factor
Reduced threat of rivalry	Differentiation	Intensifying product functionality; Intensifying product awareness; Intensifying switching cost
	Cost advantage	Credit rating; Stock price
Enhanced supplier relationship	Increased supplier	Supplier extension; Availability of company searching
	Enhanced supplier manipulation	Enhanced negotiation; Enhanced quality management
Enhanced customer relationship	Increased customer	Customer extension; Availability of customer searching
	Enhanced service	Availability of product/service information; Administrative support

6.4 Component: SISC

The key role of the SISC component is to provide the criteria for group decision making on security controls. This paper takes Leem's model to suggest the first level of criteria [44]. Lynch, Scott, CSE, and ISO9126 are used to derive the second and third level of criteria [24, 53, 54, 55]. The decision criteria are described in table 8.

Table 8. Decision criteria of SISC component

1st depth	2nd depth	3rd depth
Credibility of supplier	track record	market share, certification, relationship
	speciality	security expertise, solution lineup, best practice, offers turnkey IT security
	coverage	geographic coverage
Competitiveness of product	sales condition	price, marketing program, maintenance, support services
	architecture	hardware requirement, OS supported, source language, source code available, NOS supported,protocols supported, component model supported
	function	preventive, detective, deterrent, recovery, corrective
	performance	functionality, reliability, usability, efficiency, maintainability, portability
Continuity of service	vendor stability	financial stability, vision and experience of the management staff
	contract terms	warranty, product liability

7 Conclusion

This paper answers difficult problems that organizations face in business environments when they try to solve information security issues by suggesting the integrated methodology for information security systems. It includes the methodology architecture, integrated model of controls, process, and components. The methodology provided in this paper is compared with previous researches which have a relation to the security engineering in various aspects. This comparison is based on Leem and Leem & Kim's research which validate the contribution, significance and usefulness of the methodology [43, 56]. Table 9 summarizes the comparison results.

Table 9. Comparison of the methodology

	Objective	Functionality	Flexibility	Usability
This methodology	- Effective planning, implementation and operation of enterprise information security systems	- Security strategy planning - Selection and introduction of security controls - Evaluation of ISMS - Economic justification of security investments	- Pattern and scenario frameworks - $(CIS)^2$ model is used to provides a structured perspective with 36 cubes of security controls	- Provision of detailed process model, criteria and case studies
SSE-CMM	- Ensuring good security engineering for systems (products)	- Evaluation of security engineering practices - Customers' evaluation of a provider's security engineering capability	- None	- Provision of domain and capability
NIST handbook	- Securing computer-based resources	- Explaining important concepts, cost considerations, and interrelationships of security controls.	- None	- Provision of security activities in the computer system life cycle
BS7799	- Auditing of information security systems	- Auditing which focuses on administrative and logical controls in preventive perspective - It lacks in provision of auditing list of physical controls [15]	- None	- Provision of auditing criteria of ten categories
ISO 13335	- Risk management	- Project planning - Risk analysis - Safeguard implementation	- Provision of four kinds of risk analysis methods	- Provision of conceptual process

The contributions of this paper are summarized as follows. The first is the provision of requirements of security engineering methodology. The second is the framework which integrates methods and tools of security engineering. The third is a suggestion of the process model and components which work as the solutions or tools for ill-structured problems of information security.

References

1. Jonassen, D.H.: Using Cognitive Tools to Represent Problems. Journal of Research on Technology in Education, Vol.35, No.3 (2003)
2. Simon, H.A.: Identifying Basic Abilities Underlying Intelligent Performance on Complex Tasks. In: Resnick, L.B. (eds.): The Nature of Intelligence. LEA (1976)
3. Chi, M.T.H. Chi, and Glaser, R.: Problem Solving Ability. In: Sternberg, R.J. (eds.): Human Abilities, An Information Processing Approach. W.H. Freeman & Company (1985)

4. Jonassen, D.H.: Instructional Design Models for Well-structured and Ill-structured Problem Solving Learning Outcomes. Educational Technology, Research and Development, Vol.45, No.1 (1997)
5. Simon, H.A.: Information-Processing Theory of Human Problem Solving. In: Esters, W.K. (eds.): Handbook of Learning and Cognitive Process. LEA (1978)
6. Sinnott, J.D.: A Model for Solution of Ill-Structured Problems: Implications for Everyday and Abstract Problem Solving. In: Sinnott, J.D. (eds.): Everyday Problem Solving: Theory and Application. Praeger Publishers (1989)
7. Voss, J.F. et al.: From Representation to Decision: An Analysis of Problem Solving in International Relations. In: Sternberg, R.J. (eds.): Complex Problem Solving. LEA (1991)
8. Voss, J.F.: Learning and Transfer in Subject-matter Learning: A Problem Solving Model. International Journal of Educational Research, Vol.11 (1988)
9. Kitchner, K.S.: Cognition, Metacognition, and Epistemic Cognition: A Three-level Model of Cognitive Processing. Human Development, Vol.26 (1983)
10. Spiro, R.J. et al.: Knowledge Acquisition for Application: Cognitive Flexibility and Transfer in Complex Content Domains. In: Britton, B.C. (eds.): Executive Control Processes. LEA (1987)
11. Spiro, R.J. et al.: Cognitive Flexibility Theory: Advanced Knowledge Acquisition in Ill-Structured Domains. Center for the Study of Reading, University of Illinois (1988)
12. Choi, S.: A Study on the Methodology to Establish the Security Systems for E-business, Mater Thesis. Yonsei University (2000)
13. SEI: A Systems Engineering Capability Maturity Model, Version 2.0. Software Engineering Institute, Carnegie Mellon University (1999)
14. NIST: An Introduction to Computer Security: The NIST Handbook. NIST (1995)
15. Kim, S. et al.: An Analytic Perspective of ISO17799 ISMS. Fifth International Conference on Operations and Quantitative Management (2004)
16. ISO13335-1: Information Technology - Guidelines for the Management of IT Security - Part 1: Concepts and Models for IT Security, No. ISO/IEC TR 13335-1:1996(E). International Organization for Standardization (1996)
17. Henze, D.: IT Baseline Protection Manual. BSI (2000)
18. Rex, R.K., Charles, S.A., Houston, C.H.: Risk Analysis for Information Technology. Journal of Management Information Systems, Vol.8, No.1 (1991)
19. Ron, W.: EDP Audting: Conceptual Foundations and Practice. McGraw-Hill (1988)
20. Tudor, J.K.: Information Security Architecture: An Integrated Approach to Security in the Organization. Auerbach (2000)
21. NIST: Security Self-Assessment Guide for Information Technology Systems, NIST Special Publication 800-26. NIST (2001)
22. Gilbert, I.E.: Guide for Selecting Automated Risk Analysis Tools (SP 500-174). NIST (1989)
23. Polk, W.T., Bassham, L.E.: A Guide to the Selection of Anti-Virus Tools and Techniques(SP 800-5), NIST Special Publication. NIST (1992)
24. Lynch, G., Stenmark, I.: A Methodology for Rating Security Vendors. Gartner (1996)
25. Schweitzer, J.A.: Protecting Information in the Electronic Workplace: A Guide for Managers. Reston Publishing Company (1983)
26. Hutt, A.E.: Management's Roles in Computer Security, in Computer Security Handbook. Macmillan Publishing Company (1988)
27. Fites, P.E. et al.: Controls and Security of Computer Information Systems. Computer Science Press (1989)
28. Vallabhaneni, S.R.: CISSP Examination Textbooks. SRV Professional Publications (2000)

29. Krutz, R.L., Vines, R.D.: The CISSP Prep Guide: Mastering the Ten Domains of Computer Security. John Wiley & Sons (2001)
30. Kim, S.: Security Consultant Training Handbook. HIT (2002)
31. Firth, R. et al.: An Approach for Selecting and Specifying Tools for Information Survivability. Software Engineering Institute, Carnegie Mellon University (1998)
32. Kavanaugh, K.: Security Services: Focusing on User Needs. Gartner (2001)
33. Beall, S., Hodges, R.: Protection & Security: Software Comparison Columns. Gartner (2002)
34. Geer, D.E.: Making Choices to Show ROI. Secure Business Quarterly, Vol.1, No.2 (2001)
35. Scott, D.: Security Investment Justification and Success Factors. Gartner (1998)
36. Blakley, B.: Returns on Security Investment: An Imprecise but Necessary Calculation. Secure Business Quarterly, Vol.1, No.2 (2001)
37. Malik, W.: A Security Funding Strategy. Gartner (2001)
38. Power, R.: CSI/FBI Computer Crime and Security Survey. Computer Security Issues & Trends, Vol.8, No.1 (2002)
39. Bates, R.J.: Disaster Recovery Planning. McGraw-Hill (1991)
40. Witty, R. et al.: The Price of Information Security, Strategic Analysis Report. Gartner (2001)
41. Harris, S.: CISSP All-in-One Exam Guide, Second Edition. McGraw-Hill (2003)
42. Roper, C.A.: Risk Management for Security Professionals. Butterworth Heinemann (1999)
43. Leem, C.S. et al.: Introduction to An Integrated Methodology for Development and Implementation of Enterprise Information Systems. Proceeding of INFORMS'99 (1999)
44. Leem, C.S.: A Research on a Consulting Methodology of Enterprise Information Systems. ITR (1999)
45. Choi, J.: A Framework of the Integrated Methodology for Industrial Information Systems, Mater Thesis. Yonsei University (1998)
46. Fisher, M.A. et al.: IT Support of Single Project, Multi-project and Industry-wide Integration. Computers in Industry, Vol.35 (1998)
47. Monheit, M., Tsafrir, A.: Information Systems Architecture: a Consulting Methodology. Proceeding of the 1990 IEEE International Conference on Computer Systems and Software Engineering (1990)
48. Kim, S., Choi, S., Leem, C.S.: An Integrated Framework for Secure E-business Models and Their Implementation. Proceeding of INFORMS'99 (1999)
49. Jeon, D.: A Study on Development of TO-BE Enterprise Model for Information Strategy Planning, Master Thesis. Yonsei University (2000)
50. Kim, S., Leem, C.S.: An Information Engineering Methodology for the Security Strategy Planning. Lecture Notes in Computer Science, Vol.3043 (2004)
51. Leem, C.S., Oh, B.: Evaluation Information Strategic Planning: An Evaluation System and Its Application. Journal of Systems Integration, Vol.10, No.3 (2002)
52. Porter, M.E.: How Competitive Forces Shape Strategy. Harvard Business Review, Vol.57, Issue 2 (1979)
53. Scott, D.: Best Practices in Business Continuity Planning. Symposium/ITxpo 2002 (2002)
54. CSE: Guide to Risk Assessment and Safeguard Selection for Information Technology Systems. CSE (1996)
55. ISO9126-1: Software Engineering - Product Quality - Part 1: Quality Model, No. ISO/IEC 9126-1:2001. International Organization for Standardization (2001)
56. Leem, C.S., Kim, S.: Introduction to an Integrated Methodology for Development and Implementation of Enterprise Information Systems. Journal of System and Softwares, Vol.60 (2002)

Design and Implementation of an Ontology Algorithm for Web Documents Classification

Guiyi Wei, Jun Yu, Yun Ling, and Jun Liu

Zhejiang Gongshang University, Hangzhou, 310035, P. R. China
weiguiyi@tom.com, {yj, yling, liujun}@mail.zjgsu.edu.cn

Abstract. Traditional methods of documents classification need characteristic abstraction and classifier training. The work of collecting trainable text terms is laborious and time-consuming. Additionally, it is difficult to abstract the characteristics from Chinese documents. In order to solve the problem, this paper proposes an ontology-based approach to improve the efficiency and effectiveness of web documents classification and retrieval. Firstly, the approach establishes an ontology model based on Hownet[6] kownledge base and its method. Then, it creates ontologies for each subclass of the classification system. It uses RDFS to convert Hownet into ontology and to define the relations among ontologies. The web documents classification is performed automatically using the ontology relevance calculating algorithm. Comparing with the method of KNN[2], the results of our experiments indicate that the accuracy of ontology-based approach is close to KNN, its algorithms is more robust than KNN, and its recalling rate is better than KNN.

1 Introduction

Finding the documents that users need among all the available documents is an important issue. With the exponential growth of web information, it becomes more and more difficult to retrieve and organize the useful materials. An approach to solve this problem is to categorize documents, as in a library where the same class of books are shelved on their own bookcase. Traditionally, document categorization has been done by humans. However, there are problems with this, in that different people may categorize the same documents differently, and people working today may produce different results to people working tomorrow. The most natural solution to this problem is to use a computer to help people categorize documents consistently, and thus be able to retrieve items of interest easily. Most web sites, which offer content services, often classify the information into lots of categories. The categories are organized in an hierarchical structure. Nowadays, many simple documents retrieval systems are being supplemented with structured organizations. Traditionally, librarians use classification systems like Dewey and Library of Congress to organize vast amounts of information. Recently, web directories such as Yahoo and LookSmart have been used to classify web pages. Structured directories support browsing and search, but the manual nature of the directory compiling process makes it difficult to keep pace with the ever increasing amount of information.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 649 – 658, 2006.

The document retrieval approach in [12] attempts to parse the contents of a document. This method uses the SFC methods from Longman's Dictionary of Contemporary English. Because SFC maintains a lot of keywords from the Longman's Dictionary which are already classified, it easily categorizes documents into appropriate classes by parsing the contents of documents. However, this approach is limited in that the keywords must be classified by humans, which is a difficult task. Others methods use keywords to retrieve documents. The systems first extract keywords from documents and then assign weights to the keywords by using different approaches. There are two problems in these methods: one is how to extract keywords precisely and the other is how to decide the weight of each keyword.To solve these problems, [13] issued approachs to automatically retrieve keywords with using genetic algorithms to adapt the keyword weights.

Our work looks at the use of automatic classification methods to supplement human effort in creating structured knowledge hierarchies. A wide range of statistical and machine learning techniques have been applied to text categorization, including multivariate regression models [8], nearest neighbor classifiers [6], probabilistic Bayesian models [10], decision trees [6], neural networks [5], symbolic rule learning [1,4], and support vector machines [7,12]. These approaches all depend on having some initial labeled training data from which category models are learned. Once category models are trained, new items can be added with little or no additional human effort.

2 Related Works

Now some researchers use the methods of machine learning to classify texts. Typical approaches to Chinese text classification are SVMs [1], KNN [2] and LSA [3]. KNN and SVM have been reported as the top performing methods for English text classification [4]. To solve information heterogeneity problems, [10] proposed a metadata dictionary as an assistant mechanism for solving semantic heterogeneity based on domain ontology. It introduced an XML-based data model to manipulate and express the metadata dictionary contents. [12] use general and intuitive knowledge representation languages for indexing the content of Web documents and representing knowledge within them. The retrieval of precise information is supported by languages designed to represent semantic content. The use of Conceptual Graphs and simpler notational variants that enhance knowledge readability is advocated.

Traditional methods of documents classification have these process steps. Firstly, they create a fixed number of predefined categories. Secondly, they collect some training texts and use them to train classifier. Thirdly, they use classifier to classify the documents into a category or multi-categories. In these methods, collecting large number of training text terms is a fussy work. It must be carefully done by hand, because these terms will affect the accuracy of classification. If the predefined categories changed, these methods must collect a new set of training text terms. This will use much manpower and material resources. At the same time, most of these traditional methods haven’t considered the semantic relations of word stems. So, it is difficult to improve the accuracy of these classification methods. However, extracting characteristic terms from Chinese texts is a difficult work because there are no distinct boundaries in Chinese sentences. In order to solve these problems, this paper put forward a new method of Ontology-based web documents classification. it creates documents classification ontologies from Hownet Chinese characters knowledge base. Then, it creates ontologies for each subclass of the classification system. It uses RDFS to convert Hownet into ontology and to define the relations among ontologies. The web documents classification is performed automatically using the ontology relevance calculating algorithm.

3 Ontology-Based Web Documents Classification

3.1 HowNet Method

HowNet is an on-line common-sense knowledge base[6]. HowNet unveils inter-conceptual relations and inter-attribute relations of concepts as connoting in Chinese lexicons and their English equivalents. in paper [9], it defines knowledge as a system encompassing the varied relations amongst concepts or attributes of concepts. The relation includes: “Hypernym-Hyponym”, “synonym”, “part-whole”, “material-product”, “attribute-host”, etc. As a knowledge base, the knowledge structured by HowNet is a graph rather than a tree. HowNet attempts to construct a graph structure of its knowledge base from the inter-concept relations and inter-attribute relations. This is the fundamental distinction between HowNet and other tree-structure lexical databases.That is, if one acquires more concepts or captures more relations, one is more knowledgeable. The design of HowNet is based on its ontological view of the objective world. All physical and non-physical matters undergo a continual process of motions and changes in a specific space and time. The motions and changes are usually reflected by a change in state that in turn, is manifested by a change in the value of some attributes. The way we understand ‘attribute’ is that any object necessarily carries a set of attributes. Similarities and differences between objects are determined by the attributes they each carry. There can be no objects without attributes. For instance, human beings have natural attributes such as race, colour, gender, age, and ability to use language as well as social attributes such as nationality, class origin, job, etc.

3.2 Convert HowNet into Ontology

An ontology is the set of concepts, the definition of concepts and the relations of concepts. According to [7], the development of ontology follows an evolving prototyping life cycle. Now, there are many ontology building methodologies, but none is widely accept. Our ontology construction method is: converts Hownet to ontology and then refines it.

1. Ontology definition

$$\textit{Ontology } O := (\ meta\_info,\ Concept,\ Relation,\ Rule\ ) \tag{1}$$

Here, *meta_info* is the Meta data of *O*. It includes the name of *O*, creator, date, etc. *Concept* is the set of concepts. *Relation* is the set of relations. *Rule* is the set of rules.

2. Ontology representation
RDF and RDFS [8] have been developed by the W3C and together comprise a general-purpose knowledge representation tool that provides a neutral method for describing a resource or defining an ontology or metadata schema. RDF and RDFS provide a syntactic model and semantic structure for defining machine-processable ontologies. So, RDFS can be used to describe Ontology.

We use RDFS to convert *HowNet* into ontology. The words of *Hownet* are the *classes* (or concepts) of ontology. The “Hypernym-Hyponym” relation of *Hownet* is converted to “SubClassOf” expression. “Synonym”, “Part-Whole” and some other relations are attributes of *classes*. As depicted in figure 1, RDFS is used to describe “Part-Whole” and a *class* (or concept type). The “PartOf” notion represents the “Part-Whole” relation, and *ID* is the identity number of a *class*.

3.3 Ontology-Based Classification

The categories must be defined befor creating ontologies. The existed categories can be used too, for example, the directory of yahoo. Then, we will create ontologies for every category. The program 1 depicts the process of creating ontologies for every category.

```
......
<rdf:RDF
  xmlns=“http://localhost:8080/MyOntology#″
  xmlns:OT="&OT;"
  xmlns:rdf="&rdf;"
  xmlns:rdfs="&rdfs;"
  xmlns:a="&a;"
>
......
<rdf:Property rdf:ID="ParOf"
  OT:comment=‘This is a relation that specifies that
     the first concept is a part of the second con-
cept.’>
</rdf:Property>
......
<rdfs:Class rdf:ID="10">
  <rdfs:subClassOf rdf:resource="#31"/>
  <a: Part Of rdf:resource="#15"/>
</rdfs:Class>
......
```

Program 1. Use RDFS to define the relation of ontologies.

To improve the effectiveness, ontologies should be maintained frequently. The process of the maintenace needs the relevant knowledges of special professional domains. As more knowledges are adopted, ontologies become more effective.

After creating ontologies for every category, the next work is calculating the relevance of the ontologies. In the ontology model, the relevance is used to measure the relevant degree between the nodes in T_On and C. In general, the relevance is

```
Input: dict_On and C.
        // dict_On is the Ontology converted by Hownet.
        // C is the name of category.
Output: T_On.
        // T_On is the Ontology creating for C.
Step1: Search C or synonyms of C in dict_On.
Step2: Take the relations of dict_On as the sides of the graph,
          take the concepts of dict_On as the node of the graph.
          Then, dict_On comes into being a graph.
Step3: Take C and synonyms of C as the center nodes of
          graph, then search N nodes which around the center.
          Take N nodes and their relations as T_On.
```

Fig. 1. Process of creating pontologies for every category

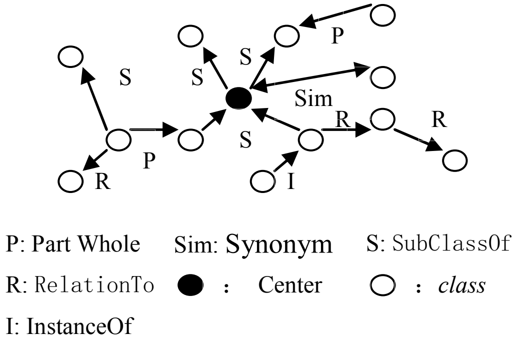


Fig. 2. An instance of T_On (as depicted in figure 2)

denoted by *relevance_score*. The figure 2 depicts an instance of T_On, which is described in figure 1.

Center is C and C's synonyms In the figure 3. According the ontology model as depicted in the figure 3 we can see that C is not an isolated concept. It locates in a relation network. In this relation network, multiple concepts (*class*) connect C, but their *relevance_score* are different. If concept C1 and C have the relation of "Synonym" and concept C2 and C have the relation of "PartWhole", C1's *relevance_score* is bigger than C2. So, the relations affect the *relevance_score*. In order to calculate the *relevance_score*, we divide the relations into four types and use *R1, R2, R3, R4* to denote them. *R1* is "Synonym" and "InstanceOf". If the weight of *R1* is w_{R1}, then w_{R1} is used to weigh the relevant degree between two concepts which have the relation of *R1*. *R2* is "Part-Whole" relation and its weight is assigned to value w_{R2}. *R3* is

“RelationTo” relation and its weight is assigned to value w_{R3}. *R4* is “subClassOf” and others relations, its weight is assigned to value w_{R4}. It suppose that the expression $1 > w_{R1}, w_{R3}, w_{R4} > 0$ is granted. The distance between the concepts and C also affect the relevance_score. If the distance is longer, the relevance_score is smaller. So, Considering the relations and distance, we can use formula (1) to calculate the relevance_score of every node in T_On.

$$Sim(t,C) = \frac{\alpha}{w\_len(t,C) + \alpha} \tag{2}$$

Here, $Sim(t,C)$ is the *relevance_score*. $w\_len(t,C)$ is the shortest distance between *t* and C. *a* is a parameter that can be used to adjust *relevance_score*. When fuction $w\_len(t,C)$ perform its calculation, it take the relations as undirected sides and set a value L_r for every side. L_r is the distance between two adjacent nodes and $L_r = 1 - w_{Ri}$. It is expressed that if two adjacent nodes have relation *Ri*, the distance of them is $1 - w_{Ri}$. The fuction $w\_len(t,C)$ can be taken as two nodes’ shortest distance. The ontology-based algorithm for web documents classification is detailedly depicted in program 2.

```
Input: the ontologies of every category and a web text
W_T
Output: the category that W_T belong to
Algorithm:
int Classify(W_T)
{
  double max_sim=0;//maximum similarity
  int index= -1;
  for (i=0; i<amount_of_category; i++)
  {
    Concept_Set=Search_concept(i,W_T);
    int number[n];
    number= Concept_frequency(Concept_Set, W_T);
    $sim(W\_T,i) = \sum_{c \in W\_T \cap O} f_{cW\_T} w_{cO}$;
    if ($Sim(W\_T,i)$> max_sim)
    {
      max_sim=$Sim(W\_T,i)$;
      index=i;
    }//end of if
  }//end of for
  return index;//return the category which W_T belongs
to.
}//end main function
```

Program 2.

4 Experiments

In the first experiment, we use KNN to classify web documents. The training text terms are taken from "CNLP Platform" (http://www.nlp.org.cn/). There is ten categories. We use 1380 texts to training and 1380 texts to test. The secong experiment uses ontology-based text classification method with the same data of the first experiment. We use precision and recall to measure the performance of these two methods.

4.1 First Test

The basic thinking of this algorithm is that given a new text, it finds out K page texts which are closest to the new text from training texts. Then decides which category the new text belongs to according to these K page texts;

1. extract characters from the training texts;
2. use feature vector to denote the training texts according to characters;
3. use feature vector to denote the new text according to characters;
4. calculates the similarity between the new text and the training texts. Then, choose K page texts which are more similar to the new text. The formula (3) is used to calculate similarity.

$$\mathrm{Sim}(d_i, d_j) = \frac{\sum_{k=1}^{M} W_{ik} \times W_{jk}}{\sqrt{(\sum_{k=1}^{M} W_{ik}^2)(\sum_{k=1}^{M} W_{jk}^2)}} \tag{3}$$

At present, there is not a good method to decide the value of K. Generally, make an initial value first, and then adjust according to the result. In the test the value of K is 35.

5. It uses formula (4) to calculate the weight of every category according to K page texts.

$$p(\vec{x}, C_j) = \sum_{\vec{d}_i \in KNN} Sim(\vec{x}, \vec{d}_i) y(\vec{d}_i, C_j) \tag{4}$$

Here, $\vec{x}$ is the feature vector of new texts. $Sim(\vec{x}, \vec{d}_i)$ is the relevance between $\vec{x}$ and $\vec{d}_i$. $y(\vec{d}_i, C_j)$ is a function. If $\vec{d}_i$ belongs to C_j, $y(\vec{d}_i, C_j)$ =1, otherwise, $y(\vec{d}_i, C_j)$ =0.

6. classify the new text to the category which has the maximal weight.

The test result is shown in table 1. Using KNN method we get average *precision*: 82% and the average *recall*: 69.1%.

Table 1. The confusion matrix of KNN. 1-politics, 2-environment, 3-computer, 4-traffic, 5-education, 6-economy, 7-military, 8-gym, 9-medicine, 10-art.

Correct	1	2	3	4	5	6	7	8	9	10
Result	250	100	100	100	110	160	120	220	100	120
1 380	243	15	2	4	9	19	66	10	7	5
2 67	0	47	1	2	2	0	0	0	14	1
3 53	0	0	52	1	0	0	0	0	0	0
4 86	0	1	1	82	0	1	0	1	0	0
5 93	1	3	0	0	84	0	2	0	3	0
6 271	2	27	41	9	9	138	1	4	18	22
7 56	0	2	3	2	0	1	44	1	3	0
8 218	1	2	0	0	1	0	7	200	5	2
9 50	1	0	0	0	0	0	0	1	48	0
10 106	2	3	0	0	5	1	0	3	2	90
precision	64%	70%	98%	95%	90%	51%	79%	92%	96%	85%
recall	97%	47%	52%	82%	76%	86%	37%	91%	48%	75%

4.2 Second Test

The Second Test use ontology-based method for web documents classification with the same data volumn of the first test.

The initial weight of R1, R2, R3, R4 is 0.9, 0.8, 0.7, 0.6. And the ontology of every category includes 350 concepts (class). The test result is: Average *precision*: 81.9%, Average *recall*: 75.8%

Table 2. The confusion matrix of ontology-based classification. 1-politics, 2-environment, 3-computer, 4-traffic, 5-education, 6-economy, 7-military, 8-gym, 9-medicine, 10-art.

Correct	1	2	3	4	5	6	7	8	9	10
Result	250	100	100	100	110	160	120	220	100	120
1 351	222	14	3	11	2	45	43	1	5	5
2 60	0	57	0	0	0	1	0	0	1	1
3 108	2	7	89	1	0	5	1	0	1	2
4 72	0	0	0	70	0	2	0	0	0	0
5 139	1	7	2	2	107	0	5	1	12	2
6 110	7	2	2	11	0	85	3	0	0	0
7 66	6	2	1	0	0	4	53	0	0	0
8 256	6	6	3	3	0	8	4	216	2	8
9 97	2	4	0	0	0	4	5	1	79	2
10 121	4	1	0	2	1	6	6	1	0	100
precision	63%	95%	82%	97%	77%	77%	80%	84%	81%	83%
recall	89%	57%	89%	70%	96%	53%	44%	98%	79%	83%

5 Performance Comparison

The figure 3 and the figure 4 depict the performance comparison of KNN method and ontology-based web documents classification method with *precision* and *recall.*

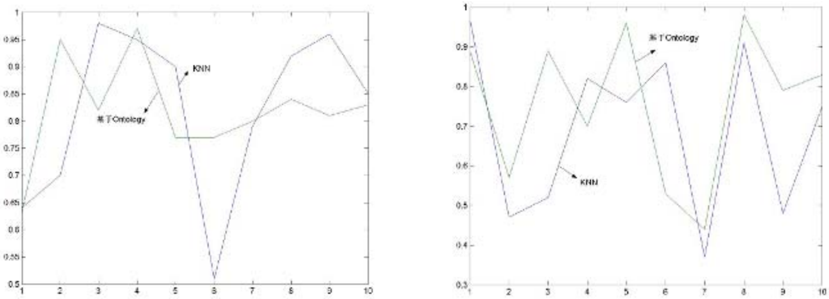


Fig. 3. Comparison of the *precision* and *recall* between KNN and Ontology-based methods

6 Conclusions

In this paper we have introduced a new approach for web documents classification. This approach doesn't need training texts. This paper proposes a ontology-based approach to classifiy web documents with using KNN's kownledge base. It supports text retrieval by keywords automatically with using an ontology algorithms to adapt their weights. The advantage of this approach is that it does not need a dictionary. This approach can retrieve any type of keywords, including types like technical keywords and product's names. The precision of document retrieval through this approach is equal to that of the PAT-tree based approach. However, the approach outlined in this paper requires less time and memory than the PAT-tree based approach does. According to chinese character documents, the performance of our ontology-based algorithms is better than genetic algorithms, including the approach proposed by KNN. The ontology-based algorithm is applied to adapt keywords' weights. The new approach is used to retrieve Chinese documents according to the weights of keywords learned.

Acknowledgments

The authors wish to thank the Natural Science Foundation of Zhejiang Province, P. R. China for the National Science Fund under grant Number Y105356. We would like to thank CRMB (the Center for Research in Modern Business, Zhejiang Gongshang University) for they partially supported this research. We would also like to thank our referees for their helpful comments and suggestions.

References

1. C. Cortes and V. Vapnik, Support vector networks, Machine learning, pp273-297, 1995(20).
2. Li Baoli, Lu Qin, Yu Shiwen, An adaptive k-nearest neighbor text categorization strategy, ACM Transactions on Asian Language Information Processing (TALIP), pp215-226, 2004.

3. Min-Yen Kan, Web page classification without the web page, Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters table of contents, pp262 – 263, 2004.
4. Marc Ehrig, Alexander Maedche, Ontology-focused crawling of Web documents, Proceedings of the 2003 ACM symposium on Applied computing, pp1174 – 1178, 2003.
5. B. Lauser, T.Wildemann, A. Poulos, F. Fisseha, J. Keizer, and S. Katz, A comprehensive framework for building multilingual domain ontologies: Creating a prototype biosecurity ontology, DC-2002: Metadata for e-Communities: Supporting Diversity and Convergence,Florence, Italy, October 2002.
6. Dong, Zhendong, Knowledge Description: What, How and Who? In: Proceedings of International Symposium on Electronic Dictionary, Tokyo, Japan, 1988.
7. Web Ontology Language (OWL), URL: http://www.w3.org/2004/OWL/ (Current November 10, 2005).
8. Resource Description Framework (RDF), URL: http://www.w3.org/RDF/ (Current November 10, 2005).
9. Report SMI-2001-0880. Stanford Knowledge Systems Laboratory. Available at www.ksl.stanford.edu/people/dlm/papers/ontologytutorial-noy-mcguinness-abstract.html
10. Ngamnij Arch-int, A semantic information gathering approach for heterogeneous information sources on WWW, Journal of Information Science, pp357, 2003.Vol.29, Iss. 5
11. Martin, Philippe, Eklund, Peter, Embedding knowledge in Web documents, Computer Networks, pp1403-1420, 1999. Vol.31.
12. Liddy, E. D., Paik, W., & Yu, E. S., Text categorization for multiple users based on semantic features from a machine-readable dictionary, ACM Transaction on Information Systems, pp278-295, 1994:12(3).
13. Chen, Lin-Chih; Luh, Cheng-Jye; Jou, Chichang, Generating page clippings from web search results using a dynamically terminated genetic algorithm, Information Systems, pp299-316, 2005.

Automatic Test Approach of Web Application for Security (AutoInspect)

Kyung Cheol Choi and Gun Ho Lee

Soongsil University, Dept. of Industrial/Information Systems Engineering,
Sangdo 5dong Dongjakku, Seoul 156-743 Korea
aackc@paran.com
ghlee@ssu.ac.kr

Abstract. We present an automatic test approach to improve the security of web application, which detects vulnerable spots based on black box test through three phases of craw, test, and report. The test process considers a blind point for security through the development life cycle, the faults of web application and server setup in a various point of attackers, etc. The test approach is applied to the web applications in industry, analyzed, and compared with the existing test tool.

1 Introduction

Contemporary, a vast majority of security intrusions are made in awareness or unawareness. Although the security of web application is getting importance, the researches on test methods for the web application have been mainly focused on the other quality factors and security has not been thoroughly accessed, which in turn a system may be vulnerable to an invader. Web applications are getting large and complex by linking or integrating a variety of web applications for a business purpose.

The existing security solution has a limitation in protecting a number of intrusions for deletion, manipulation, and stealth of important personal, enterprise, or web site information. Even though a fire-well doesn't allow undesired port for request, it doesn't detect data manipulations in the allowed port. Since an invader detection system provides patterns based detection roles, it isn't able to detect anomalistic attacks like cookie poisoning and parameter tempering. The attackers like to dig out backup sources, any comments, etc. which residue in the web system after the web application development.

One way to reduce vulnerability of web application is to consider security problem in the development life cycle [9]. At test stage, automatic tools based on efficient approaches are necessary to scrutinize the web application for vulnerability. Web applications for security have not been studied relatively in academic and a few papers on test have been published in functional aspects [19] but no research paper on web application security tool has been shown yet. We present Automatic test approach of Web application for Security (AutoInspect) based on black box test and assesses of vulnerability of a variety of web applications in industry and compared with an existing tool, ScanDo [14] in the detection ratio and in the running time.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 659 – 668, 2006.

Since a tester may or may not know the inside of details of a web application under test, the tester considers a testing process in a black box test or in a white box test [8]. Black box test method is a technique for hardening and penetration testing of web applications where the source code to the application is not available to the tester. The tester identifies the response of the application by putting white space, SQL keywords, over sized requests to a parameter, and cross site scripting, which may be automatically performed. Should the messages of Internal 'Server Error 500' or half-loaded pages show up, it means the application is vulnerable for an intrusion [15].

AppScan[2], Achilles[1], and ScanDo [14] are available tools today to test vulnerability of web application. ScanDo is a comprehensive vulnerability assessment scanner from Kavado Inc. that audits the entire web application environment but it can be run on a few Microsoft Windows platforms. ScanDo scans the web application three phases: First at crawl phase, it spiders through the application and parsers data for a testing purpose, while at the same time registering the contents and structure of the application. It then goes through web pages and tries to determine which parts of the application are vulnerable and to perform actual attacks on the pages. Lastly it collects the results of the scan into a report [19]. AppScan of Sanctum Inc. runs as a stand-alone application on a workstation machine and does its scanning from an external viewpoint which means it doesn't utilize any inside knowledge of how the tested web application is built. AppScan has a similar process with ScanDo that the application is evaluated first, tested, and lastly the result are reported. Human intervention is only needed in the first and third part of process. WebInspect [17] is business-oriented, user-friendly interface, based on a database of vulnerability signatures as well as some artificial intelligence, and similarly runs with ScanDo. OWASP provides Webscrab as a freeware scanner that is a similar with other commercial tools [18]. Achilles [1] is a proxy server, which is used as a test tool to manipulate client script information and cookie.

Since there are no evaluation versions or open-sources of Appscan and WebInspect, we compare AutoInspect and ScanDo.

2 Test Items and Detection Ratio of Vulnerability

In this study, we use 10 test items similar to the items of the literatures [13] and interview 62 persons who are responsible for web application security to obtain the weight of each test item based on nine intensities of importance. The 10 test items are rated for a priority by using Saaty's analytical hierarchical process method [6][10] in Table 1. We classify three categories of security problems, i.e., authentication, disclosure, and tampering. Authentication includes injecting SQL queries and cookie poisoning. Disclosure includes unnecessary HTTP method, directory browsing, sample and test files, backup files, vulnerability patterns, and disclosure of personal information. Client side script and parameters are tampered.

To show a consistency of the weights of 10 test items from 62 interviewees, we compute Saaty's Consistency Index (CI) [10]. Priority O_{ij} is calculated as c_j/c_i, the relative importance of the category j compared with i where c_i and c_j are a weights of categories i and j, respectively. The normalized priority of category i, $R_i = \sum_{i=1}^{3}(O_{ij} / \sum_{i=1}^{3} O_{ij})$. With similar way, priority $P_{st} = w_t/w_s$, the relative importance of the test items t compared with s where w_s and w_t are the weights of row test item s, and column test item t in a pair wise matrix of a category. The normalized priority of test item t of category i , $Q_{it} = \sum_{t\in zi} (P_{st} / \sum_{s\in zi} P_{st})$, s, $t \in Z_i$ where Z_i is a set of test items in category i. Therefore, we can obtain the scaled priorities of test item t of category i, $V_{it} = R_i Q_{it}$, $t \in Z_i$.

Saaty's CI = (λ_{max} -n)/(n-1) where $\lambda_{max} = \sum_{i=1}^{3} (\sum_{j=1}^{3} O_{ij} R_j) / R_i$ and n is the number of test categories. We can see that the result in Table 1 for the three categories is reliable since Saaty's CI, i.e., (0.019) < 0.1 is satisfied from λ_{max} = 3.038. With similar way, we obtain λ_{max} =2 and CI = 0 from the test items of authentication, λ_{max} = 6.10 and CI = 0.02 from the test items of disclosure, and λ_{max} =2 and CI = 0 from the test items of tampering, respectively where CI =0 means that the priority values of test items of tampering are perfectly consistent. Therefore, the weights of 10 test items from 62 interviewees show a consistency. For more information on CI, see Saaty's paper [10].

We formulate a detection ratio of vulnerability in a web application testing. To formulate a detection ratio of vulnerability, D for a web application in (1), let μ_t be one if test item t is detected and else zero.

$$D = (\sum_{i=1}^{3} \sum_{t\in Zi} V_{it} \mu_{it} / \sum_{i=1}^{3} \sum_{t\in Zi} V_t) \times 100(\%) \tag{1}$$

3 Test Process for Security

The test items in Table 1 are similar to those of the existing tools but a detecting method in each test item is different from each other, which determines the detection ratio of the tools. We present two test methods to assess a web application for vulnerability: The one uses fault injection method as a replacement method, i.e., SQL injection and parameter tampering, and a pattern injection method as an appending method, i.e., HTTP option disclosure, directory browsing, sample and test files, backup files, and vulnerability patterns, using pattern DB which is prepared for an automatic process. The other is a method of automatic collection of the location of a client script page, which requires logic analysis and manipulation of source code about the client script [4].

Table 1. Priority of Test Item

Item #	Test item	Q_{it}	V_{it}	Category	R_i
A1	Injecting SQL queries	0.50	0.95	Authentication (A)	1.91
A2	Cookie poisoning	1.50	2.85		
D3	HTTP Option Disclosure	0.46	0.14	Disclosure (D)	0.32
D4	Directory Browsing	1.34	0.42		
D5	Sample and Test files	0.30	0.09		
D6	Backup Files	0.65	0.20		
D7	Vulnerability Patterns	0.90	0.28		
D8	Personal Information	1.90	0.59		
T1	Client Side Script Tampering	0.50	0.39	Tampering (T)	0.78
T2	Parameter Modulation	1.50	1.17		

The test procedure of web application includes three phases, crawl, test, and report. At crawl phase, it collects the structure, contents, and environment of web application., i.e., it parsers codes of HTML, flash object, JavaScript, and VBscript and collects information of URL and personal information such as social ID #, credit card #, phone # , etc. At test phase, the test patterns are transferred to web application using the information collected at first phase, i.e., it detects SQL injection, cross-site scripting, parameter tampering, hidden field manipulation, cookie poisoning, stealth commanding, and disclosure of personal information. At report phase, it reports the vulnerable parts as results of the assessment.

We also use automatic single pattern to test HTTP option disclosure, directory browsing, common files, backup files, and vulnerability patterns, respectively. Automatic composition pattern is used to test injecting SQL queries and parameter manipulation. Client logic analysis is used for client side script tampering and cookie poisoning.

3.1 Single Pattern Transport Type

Single pattern transport type, based on test pattern DB, finds some vulnerability by sending the pattern to web application and analyzing server reply code.

HTTP Option Disclosure (D1). We send OPTION, TRACE, PUT, and DELETE, respectively of HTTP methods to application directory and analyze server reply code since OPTION provides all available HTTP methods, TRACE allows to insert script type pattern and attacks cross site scripting, PUT allows to upload remotely some web contents, and DELETE allows to delete remotely some web contents.

Directory Disclosure (D2). We investigate '/XXX' directory disclosure if '<title>Index of /XXX </title>' is found in HTTP reply page.

Disclosure of Sample and Test files (D3). We transport a pattern that doesn't exist in a web system and then the message of '404 NOT FOUND' or the message in the system should be appeared, otherwise we regard it as the vulnerable web system.

Backup Disclosure (D4). We investigate file extensions related to backup left over in a web system for backup source code by a programmer or a development tool.

Vulnerability Patterns Disclosure (D5). We investigate web application setup, and files and CGI (Common Gateway Interface) pattern.

Personal Information Disclosure (D6). We use a regular expression which is a pattern of characters that describes a set of strings like social security #, credit card #, mobile phone #, etc.

3.2 Composition Pattern Transport Type

Composition pattern transport based on the collected information from web site sends composition patterns to the web site and analyzes server reply code to find a vulnerable part.

SQL Injection (A1). An attacker is able to insert a series of statements into a query, manipulating input data of web application. If the input data is concatenated as a part of a SQL query, the special characters can be used to construct a custom SQL query. We try to run SQL query using the input data and complete SQL query in dynamic SQL query module or use the assigned input data to the variable. SQL injection may lead to leak some information, insert wrong information, maliciously modify data, or delete the database [3][17].

GET type in pattern transport inserts a pattern(s) into each variable and then test the vulnerability. The system is invulnerable if a message of ‘Password is not entered’ or a message in the system is shown for each pattern of

http://xxx.xxx.xxx/logi n.asp?id=‘-- &pwd= .

A web application is invulnerable if a message of ‘Account Information is not entered’ or a message in the system is shown for each pattern of

http://xxx.xxx.xxx/login.asp?id=&pwd =‘---.

A web application is invulnerable if ‘Internal Server Error 500’ message is shown for each pattern of

http://xxx.xxx.xxx/logi n.asp?id=‘-- &pwd= ,
http://xxx.xxx.xxx/login.asp?id=‘--&pwd=’--.

A web application is invulnerable if ‘Internal Server Error 500’ message or a message in the system is shown for a pattern of

http://xxx.xxx.xxx/login.asp?id=‘--&pwd=’--.

POST type in pattern transport inserts a pattern(s) into each variable and then tests the vulnerability. For example, a web application is invulnerable if message of ‘Password is not entered’ or an error message in the system is shown for the pattern of

http://xxx.xxx.xxx/login.aspid='-- pwd=.

A web application is vulnerable if the error message of ‘Internal Server Error 500’ is shown for the pattern of

http://xxx.xxx.xxx/login.aspid='—pwd=.

With similar way, we try to for each pattern of

http://xxx.xxx.xxx/login.aspid=pwd='--,
http://xxx.xxx.xxx/login.aspid='-- pwd='--,
http:// xxx. xxx.xxx/login.asp id='-- pwd='--.

Since the pattern transport type inserts a pattern(s) into a variable, which transports from a web page to server, it sends the pattern by passing the web page, which has java validation process. Since it may not be able to test the vulnerability from the user side script, we insert the pattern(s) into the sending data rather than insert the pattern into an input form of the web page. Hence, we could test vulnerability only when it is filtered in server side.

```
Function of test for user input using java script
  function Go_Login(){
    if(document.Login_form.User_ID.value.length < 2){
      alert('Input ID.');
      document.Login_form.User_ID.focus();
      document.Login_form.User_ID.select();
      return;
  }
  if(document.Login_form.User_Pass.value.length < 2){
    alert( 'Input Password.');
    document.Login_form.User_Pass.focus();
    document.Login_form.User_Pass.select();
    return;
  }
 document.Login_form.submit();
}
```

Parameter Manipulation (T2). Parameter manipulation is similar to cross site scripting [7]. When a user submits information to the web application through some mechanism, the information can be disclosed to other users. Hence, it is possible for an invader to insert malicious HTML and script into the information. This also leads the user to click URL of an email or web board hiding flash or malicious program. Cookies or session information which include a user login ID and password will be passed to attacker at moment that a user opens a web board or an email where flash or malicious program is hiding.

Since HTTP has protocol stateless, which doesn't store session, cookie and session are necessary to collect and store user's information including user's password and account ID which are issued after authentication process. The information of cookie or session may be used for any purpose by stealth without any real information about the user [7].

Pattern Transport Types is able to identify whether parameters used in web application are vulnerable or invulnerable by inserting a vulnerable pattern into each parameter in sequence. For example, the system is invulnerable if the web page is not changed or an error message in the system is shown for each pattern of

http://xxx.xxx.xxx/servlet/View?a=<script>alert("test")</script>&b=182&c=2,
http://xxx.xxx.xxx/servlet/View?a= bbs&b=<script>alert("test")</script>&c=2,
http://xxx.xxx.xxx/servlet/View?A=bbs&b=182&c=<script>alert("test")</script>.

A web application is invulnerable if Internal Server Error 500 or the java script inserted is shown for each pattern of

http://xxx.xxx.xxx/servlet/View?a=<script>alert("test")</script>&b=182&c=2,
http://xxx.xxx.xxx/servlet/View?a=bbs&b=<script>alert("test")</script>&c=2,
http://xxx.xxx.xxx/servle t/View?a=bbs&b=182&c=<script>alert("test")</script>,

Tester should analyze and manipulate source codes of cookies, session, client script authentication, client script, and hidden field since an attacker can easily detect, collect, recognize, and cook the source codes.

3.3 Cookies (Session) Manipulation

Web application establishes cookie or sessions to keep track of the stream of requests between client and server but HTTP doesn't provide this function since HTTP protocol is stateless. Web applications create it by programming method like user side cookie or server side session. If cookie or session tokens are not properly protected, an attacker can modify or hijack a user's session and get the user's personal information.

Client Script Authentication Manipulation. Since it is possible to download client script sources through web browser, client script sources may be divulged. For example, see the source code of bulletin board for deletion next. An invader may delete or manipulate the codes of bulletin board since authentication process of java script is used instead of cookie or server side session.

```
Authentication Structure of java script
  <script>
     UserID =        "";
       UserType      =       "";
       UserName      =       "";
```

```
    UserIP        =      "";
    UserSessionID =  "76ff2c954bae5e62b5f52bd251bb8464";
</scrip>

<!—Delete e-board by java authentication -->
Function DeleteDetail(Usertype){
  if(Usertype != '3'){     .
     if ('10' != UserID){       .
       alert('User ID can’t delete this message.');
            return;
     }
     else{
       if(!confirm('Delete ?')) return;
       ExecuteDelete();
     }
  }
}
```

Client Script Manipulation identifies authentication of user input data by detecting user’s general information or user’s business information using client script. For example, see SSN identifying code using java script. Since the server may not detect data sent by user, the user can manipulate the information on client side.

```
SSN identifying code using java script
  function input check(){
    if(document.inForm.jmno1.value == ""){
       alert("Input Social Security Number.");
       document.inForm.jmno1.focus();
       return;
    }
    if(document.inForm.jmno2.value == ""){
        alert("Input Social Security Number.");
        document.inForm.jmno2.focus();
        return;
    }
    var chk =0;
    var ResNo1 = document.inForm.jmno1.value;
    var ResNo2 = document.inForm.jmno2.value;
    for (var i = 0; i <=5 ; i++){
    chk = chk + ((i%8+2) * parseInt(ResNo1.substring(i,i+1)));
    }

   for (var i= 6 ; i<=11 ; i++){
    chk = chk + ((i%8+2) * parseInt(ResNo2.substring(i-6,i-5)));
   }
   chk = 11 - (chk % 11);
   if (chk != 0){
    alert ("Social Security No is Not Valid.");
    document.inForm.jmno1.focus();
    return;
  }
}// end of function
```

3.4 Hidden Field Manipulation

The client side script of hidden field has been used for product information like price, e.g.,'<input type="hidden" name="product110AA" value="109.99" >'. Since the hidden field is downloaded and the source code of web browser can be opened like HTML, a malicious user can manipulate the information, which is transported to server at final purchase stage.

4 Analysis and Comparison

Since evaluation versions and source codes of Appscan and WebInspect are not available, we evaluate the two tools based on demo version and a literature. In this section, we construct ScanDo tool based on source code and compare detection ratios of AutoInspect and ScanDo by applying to 18 web applications in Table 3. Although JavaScript is widely used in a numerous web applications, the three tools are not able to parse any code, which means that the three tools can't detect any vulnerable spot. The existing three tools represent the importance of the test items to three discrete levels, i.e., low, medium, and high, which may not consider sensitiveness between the adjacent levels. The test process presented provides the relative importance index of detected items, which stand for a priority among the test items

We select substantial 18 web applications, set up error by manually or by test tools and run ScanDo and AutoInspect for each web application. It shows 29.73% and 49.56 % of arithmetical averages of detection ratios in ScanDo and in AutoInspect, respectively.

AutoInspect doesn't detect test item T1 in web application 5 since the developer uses the type of form control, which is defined by INPUT and given by the TYPE attribute and that field works as an object representation of a hidden field in an HTML form. AutoInspect doesn't identify test items D2, D3 in web applications 10 and 11 since the related patterns are not included in pattern DB at the time. Therefore, patterns in the database and organization of pattern database affect on detecting items in a reasonable time.

Detection times of two tools are compared in Fig. 1. AutoInspect runs less than 40 CPU minutes of arithmetical average to assess web applications whereas it takes 300 CPU minutes of arithmetical average for ScanDo to assess web applications. AutoInspect runs faster as 7.5 times as ScanDo since the parser engine of ScanDo probably is not able to parse html or anything else properly.

Table 2. Comparison of Test Tools

Test Tool	Parsing	Detected Item Representation
ScanDo	Not available in JavaScript	Three discrete levels
Appscan	Not available in JavaScript	Three discrete levels
WebInspect	Not available in JavaScript	Three discrete levels
This study	Available in JavaScript	Scaled priority numbers

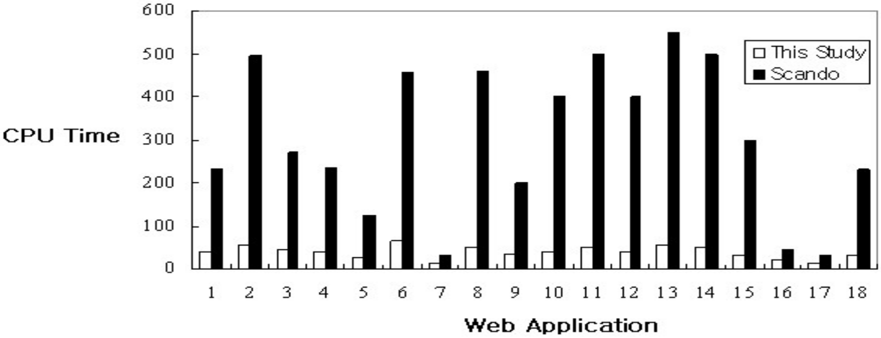


Fig. 1. Comparison of detection times (minute)

Table 3. Detection Ratio (D)(%)

M: Manual, P: Penetration,, T: Test Tool.

No of web application	ScanDo		AutoInspect		Error Setup
	Items detected	D	Items detected	D	
1	A1,A2,T1,T2	76	A1,A2,D1,D3,D4,D5,T1,T2	86	T
2	T2	17	D1,D2,D3,D4,D5,T2	32	T
3	A1,A2,T1	59	A1,A2,D3,D4,D5,T1	72	T
4	A1,T1,T2	36	A1,A2,D3,D4,D5,T1,T2	84	T
5	A1,T1,T2	36	A1,D3,D4,D5,T2	38	M
6	T1	6	D1,D2,D3,D5,D7	15	P
7	T2	17	D1,D3,D4,D5,T2	27	T
8	A1,D4	16	A1,D3,D4,D5,T2	38	T
9	A1,A2, T1	59	A1,A2,T1	59	T
10	D4	1	A1,D3	15	T
11	D4,D5	7	A1,D3,D5,T2	43	T
12	A1,D4,D5	20	A1,D3,D4,D5,T2	38	T
13	A1	13	A1,A2,D3,D4,D5,T2	78	M
14	D1,D2,D4,D5,D6	23	D1,D2,D3,D4,D5,D6	24	M
15	D1,D2,D4,D5,D6	22	D1,D2,D3,D4,D5,D6	24	M
16	A1,D1	15	A1,A2,D1,D3,D4,D5,T2	80	M
17	A1,A2,D5	58	A1,A2,D3,D4,D5,T2	78	T
18	A1,A2	54	A1,A2,T3,T4,T5	61	T
Average of D (%)		29.7		49.6	

5 Conclusion

A large and complex of web application and rapid development phases with extremely short turnaround time making it difficult to eliminate vulnerabilities and to improve security quality. It is impossible to test automatically all parts of a web application since human tester should necessarily analyze client side script tampering and cookie poisoning. It is meaningful that the method of AutoInspect is based on numerous experiences of automatic and manual web application tests. Project management technique of the development for security, security concerns of developers or management, and proper testing are important factors to improve security in a web application.

Security issues on web applications are a relatively new field to academic study. Security is an important issue as much as function in a management of web application. More researches should be performed on the assessment of web application for security rather than depending on the only industrial experiences.

Acknowledgements. This research is funded by Soongsil University.

References

1. Achilles: Web Application Proxy Tool. http://www.owasp.org
2. Appscan: Web Application Testing Tool. http://www.watchfire.com
3. Arkin, B., Stender, S., McGraw, G.: Software Penetration Testing. IEEE Security & Privacy, Vol. 3, No.1. (2005) 84-87
4. AppsecInc: Manipulating Microsoft SQL Server Using SQL Injection. http://www.appsec Inc.com/presentations/Manipulating_SQL_Server_Using_SQL_Injection.pdf, (2002)
5. Auronen, L.: Tool-Based Approach to Assessing Web Application Security. Helsinki University of Technology, (2002)
6. Borgelt, C., Kruse, R.: Induction of Association Rules: Apriori Implementation. 15th Conference on Computational Statistics Compstat, Berlin, Germany (2002)
7. CgiSecurity: CA-2000-02 Malicious HTML Tags Embedded in Client Web Requests. http:// www.cgisecurity.com/articles/xss-faq.shtml, (2002)
8. Ghosh, A.K., McGraw, G.: An Approach for Certifying Security in Software Components. Proceedings of the 21st National Information Systems Security Conference, October 5-8, Crystal City, VA.(1998)
9. Heineman, K.: Building Web Application Security into Your Development Process. http://www.spidynamics.com/whitepapers/Webapp_Dev_Process.pdf, (2003)
10. Multi-criterion decision-making. http://ecolu-info.unige.ch/~dubois/Mutate_final/Lectures/ Lect131/lect131.htm
11. Noriyuki, M., Ken, N.: Interactive Support for Decision Making. Institute Policy and Planning Sciences, Univ. of Tsukuba, Nissan Motor, Co. Ltd. Nissan Technical Center
12. NGS Software: Advanced SQL Injection In SQL Server Applications. http://www.nextge nss.com/papers/advanced_sql_injection.pdf, (2002)
13. OWASP: Top 10 Most Critical Web Application Security Vulnerabilities. http://www.owas p.org/documentation/topten.html (2004)
14. Scando: Web Application Testing Tool. http://www.kavado.com
15. SecurityFocus: Black Box Test Method. http://www.securityfocus.com/infocus/1709
16. Hoo, K.S., Sudbury, A.W., Jaquith, A.R.: Tangible ROI through Secure Software Engineering. Secure Business Quarterly, Vol. 1. No. 2 (2001)
17. WebInspect: Web Application Testing Tool. http://www.spidynamics.com
18. WebScrab: Web Application Testing Tool. http://www.owasp.org
19. Wen, Y., Kun, S. , Lin, T.P.: Web Application Security Assessment by Fault Injection and Behavior Monitoring. The 12th International W3 Conference 20-24 May, Budapest, HUNGARY (2003)

A Scenario-Based User-Oriented Integrated Architecture for Supporting Interoperability Among Heterogeneous Home Network Middlewares*

Min Chan Kim[1] and Sung Jo Kim[2]

[1] Dept of Computer Science & Engineering, Chung-Ang University, 221 Huksuk-Dong,
Dongjak-Gu, Seoul, 156-756 KOREA
barrios@konan.cse.cau.ac.kr
[2] Dept of Computer Science & Engineering, Chung-Ang University, 221 Huksuk-Dong,
Dongjak-Gu, Seoul, 156-756 KOREA
sjkim@cau.ac.kr

Abstract. There exist many home network middlewares such as Havi, Jini, LonWorks, UPnP, and SLP for the purpose of the information appliance control. As home networks evolve, new middlewares specialized for diverse information appliances will appear continuously. In this paper, we examine an integrated architecture for supporting interoperability among heterogeneous home network middlewares and present a scenario-based user-oriented integrated architecture for home automation, which controls and interoperates information appliances by integrating heterogeneous home network middlewares with an ability of reflecting flexible properties of home network middlewares.

1 Introduction

The evolution of high speed communication, the Internet, the digital hardware technology has promoted the intelligence of information appliances. This has led to the development of middlewares such as UPnP[1], Jini[2], Havi[3], LonWorks[4]. SLP[5], etc. supporting service discovery and interaction that simplify the installation of information appliances with an ability to discover other services(e.g., application softwares and devices which can be used by other applications or services accessing them) dynamically.

However, owing to the heterogeneity of home network middlewares, service application developers must either develop applications for each appliance which provides the same services reflecting properties of each middleware, or develop applications large enough to support each middleware simultaneously. We anticipate that the heterogeneity of home network middlewares will be continued and such heterogeneous services and protocols will be mixed in the future service environment[6][7].

Therefore, the need for solutions to the heterogeneity of middlewares is great in the field of home network research. Moreover, mechanisms for supporting interoperation

* This research was supported by the MIC(Ministry of Information and Communication), Korea, under the Chung-Ang University HNRC-ITRC(Home Network Research Center) support program supervised by the IITA(Institute of Information Technology Assessment).

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 669 – 678, 2006.

services should be invented so that home users can use various home network devices efficiently. The answer to the question, “Why do we try to integrate diverse middlewares of home network?”, is to interoperate existing diverse middlewares. Then, what the question “Why is interoperability necessary?” contributes is to provide of the convenience of life to users by implementing home automation through the operation of heterogeneous home network appliances, which is the purpose of our research as well.

In order to provide various necessary services for home automation, it is essential to interoperate heterogeneous middlewares. For the interoperation services, different services should be executed consecutively with priority given to a specific entry point. This entry point can have a variety of formats. A specific event occurs with a specific context, and a specific scenario is executed starting from that event. The next scenario is executed based on the result of that executed scenario and the context at that point and appliances of home network cooperate and provide a new service.

This paper is organized as follows. In Section 2, we describe the heterogeneity for supporting interoperability among appliances under heterogeneous home network, introduce researches in progress for the development of integrated middlewares, which can resolve the heterogeneity issues, support interoperation of heterogeneous home network middlewares, which can be remotely controlled, and analyze problems caused by them. In Section 3, we present a framework developed in this paper, and in Section 4, we show the result of the framework implemented by us. Finally, in Section 5, we conclude the paper and describe the future work.

2 Related Works

In the home network environment, new formats of appliances are continuously developed, services are too divers to be standardized, and predictions of the advent of these appliances can not be easily made. Also, it is much more difficult to predict the future since the existing services and appliances would be modified easily. Even though it is predictable, there is a limit to solving fundamental problems. These problems are brought at a semantic stage. It is not simple at all to solve and even occasionally to recognize the problems. We will describe the details of semantic problems in Section 2.2. Also, syntax problems caused by the difference between the service discovery and invocation mechanism should be solved for interoperability.

2.1 Interoperation Mechanisms of Heterogeneous Home Network Middlewares

Interoperation mechanisms of heterogeneous home network middlewares, currently under investigation, are classified into two types, bridge protocol and an integrated framework. .

The former provides a way to support interoperability between heterogeneous middlewares. UPnP-to-Havi bridge[9] was developed by Thomson Multimedia and Phillips, and the interoperation of Jini and UPnP[10] has been researched at the University of New Orleans. Being useful for the interoperation between two specific middlewares, it has a scalability problem since it fails to provide consistent way for interoperation of various types of middlewares as the number of brides can be increased and connections can be complicated as well with the advent of new middlewares.

The latter provides an abstracted common layer above various middlewares and has an architecture that bridges each middleware based on the common layer. This type of architecture has an advantage that, even though new middlewares are developed, it can be easily integrated with other middlewares if an appropriate agent is implemented. At Waseda University, middleware integration[12] has been tried through SOAP(Simple Object Access Protocol)[11] gateway configuration and researches on interoperation services among heterogeneous middlewares have been under way at OSGi Alliance[13] and ETRI[14]. However, there are several problems with the interoperation mechanism that bridges one interface with the interfaces of multiple middlewares. These problems will be discussed in Section 2.2.

2.2 Analysis of Integrated Framework-Based Mechanism

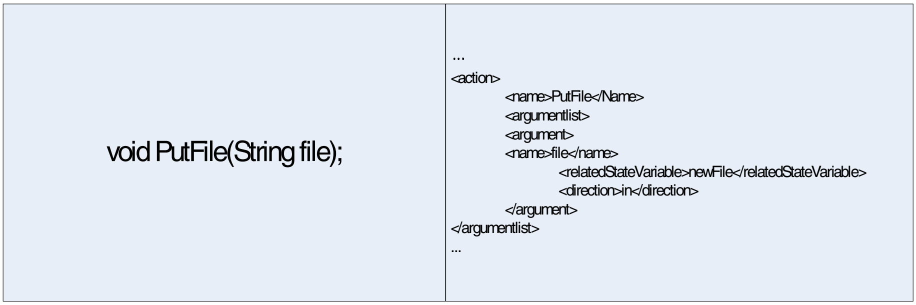
A model presented in this paper is based on an integrated framework. We will address four issues which should be considered when designing an integrated framework-based mechanism in this section.

1) How can devices adopting different middlewares find each other transparently?
Each middleware utilizes different service discovery mechanisms. Due to the differences between different mechanisms, even devices providing compatible services cannot recognize each other if each adopts heterogeneous middlewares.

2) How can services adopting different middlewares invoke each other?
A service invocation mechanism of Jini relies on Java RMI(Remote Method Invoke) which uses Java byte code. UPnP use SOAP(Simple Object Access Protocol) to invoke a service by transferring XML text stream. Problems caused by the difference between service invocation mechanisms must be solved to provide interoperability between heterogeneous middlewares. In order to solve the problem, syntax elements of middlewares(e.g., method name, the order or arguments, the type size of return value) should be adjusted. It is also necessary to convert calling method according to service invoke mechanisms of each middleware.

3) How can the integrated framework recognize that heterogeneous services are mutually interoperable?
If service interfaces(i.e. syntax elements) are identical, can we assume that interoperable services are provided? For example, suppose that a method provided by a Jini's storage service is as shown in Fig 1-a) and the provided service stores new data in a file. Similarly, suppose that a UPnP storage service is as shown in Fig 1-b) and it has a PutFile method in SCPD(Service Control Protocol).

However, the UPnP storage service requires a user authentication process and new data can be stored in a file only after passing the process. If the authentication fails, it returns an error. In this case, syntax elements look identical, but, because of the difference between semantic elements, a method to adjust these elements is necessary for interoperability.



```
void PutFile(String file);
```

```
...
<action>
          <name>PutFile</Name>
          <argumentlist>
          <argument>
          <name>file</name>
                    <relatedStateVariable>newFile</relatedStateVariable>
                    <direction>in</direction>
          </argument>
</argumentlist>
...
```

Fig. 1. Storage methods of Jini and UPnP and SCPD

A typical way to provide interoperability among services which are syntactically identical syntax components but semantically different is to define a standard interface for integrated midddlewares for each device(e.g., TV, MP3, Printer, etc.) and to use table-formatted translation bridges for each middleware according to the definitions. However, static bridging to such a single standard interface has a limit to reflect dynamic properties of home network middleware protocols whenever new functions or devices are inserted[15].

Moreover, whenever new devices appear it is very time consuming to define standards and adjust to those standards as we have experienced previously. Instead, application developers should develop applications which can understand all the services under different middlewares whenever they develop services, but it is impossible in reality because, among these interfaces, new interfaces can continue to appear after applications are developed, Moreover, it takes too much time to define standards and adjust to those standards whenever new devices are added.

4) Is interoperation services required by users applied to the home network environment immediately?

According to the investigation conducted by KISDI (Korea Information Strategy Development Institute)[16], home network usage pattern of users can be classified into data network, entertainment, and home automation and the number of users who want to use the home automation is not few. In spite that the potential market size is very large, which there are many ongoing researches in areas of entertainment and data network, there have been few researches on middlewares for supporting home automation. For supporting home automation under diverse middlewares, the various techniques for interoperability among heterogeneous middlewares must be considered before anything else. Furthermore, users should be able to modify interworking scenarios with the techniques whenever they want. Consequently, an application with the static architecture which provide only the static scenarios do not meet this kind of requirement.

In this paper, we design a framework that guarantees interoperability among home network services based on the home automation and reflects user's requests immediately.

3 HOMI Architecture Design

HOMI is a scenario-based user-oriented integrated framework designed in consideration of service usage patterns. Instead of using a mechanism that bridges and integrates interfaces utilizing a single standard interface to solve semantic problems caused by the difference of heterogeneous middleware interfaces during the interoperation process, HOMI uses a method that provides an easy and simple interpreter language to users so that interworking scenarios can be directly designed by the user in order to verify flexibility of interoperation and accuracy of compatible interface selection.

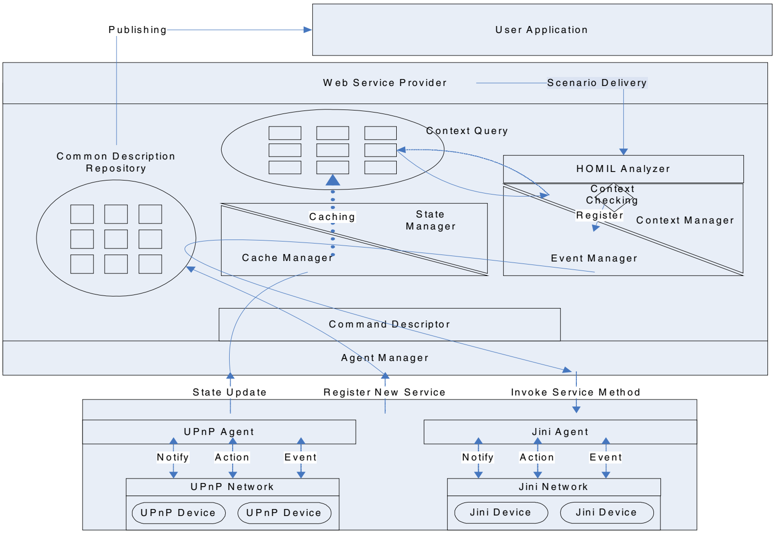


Fig. 2. HOMI architecture

HOMI provides an interpretive language called HOMIL(HOMI Language) which support to design a preferred interworking scenario in a simple way to end users. HOMI consists of 5 major parts; Agent Manager, State Manager, HOMIL Analyzer, Cache Manager, Context/Event Manager.

3.1 Agent Manager

It has been already mentioned previously that the home network middlewares for the operation of home appliances cannot be interoperable due to different protocols and execution mechanisms. In order to integrate these heterogeneous home network middlewares into one framework, a method to abstract different middlewares into one is necessary.

For this, we describe an abstract layer called Common Description in this paper. This layer consists of components essential to home appliances in order to contain all the common parts of diverse middlewares. Appliances with an ability to operate under home network environment must have at least three components; Service Description, Service Method(Action), Service State.

Service description is a service specification which can be understood by human beings and Service Method is a function performed by a service. Service State is a value that represents the status of appliances. Appliances must provide a mechanism that notifies its status to the outside or assists the outside to recognize the status. It is possible for an application to interoperate with other appliance as long as this kind of mechanism is provided either from the outside or inside of the appliance. UPnP and Jini either support directly this kind of mechanism or have an architecture which can support this mechanism.

3.2 State Manager

In order to integrate and interoperate appliances on the home network and to trigger interworking scenarios, the status of appliances is very important.

When the status is changed after a particular service is executed, it acts as an event to trigger the next scenario. In order to solve problems caused by the difference between state advertisement mechanisms[1][2], each agent monitors the status of appliances under its control and notifies State Manager of HOMI of the result. The sate advertisements of UPnP adopts event-based Publisher/Subscriber scheme. Therefore, when a new device is found, UPnP agent requires a subscription to the device to monitor the status of the discovered device and receives sid(subscription identifier) as a result. If the subscription request is successful, UPnP agent is notified of changed status of the subscribed device and sends the result back to HOMI State Manager.

In contrast, there have no standard mechanism to transfer status in Jini. To solve this kind of problem, Jini makes it possible to notify status change of a device status using attributes provided by Jini. HOMI is notified of status changes through events that occur when the service attributes change. Like this, Jini still lacks the specification to be used as a home network middleware in many aspects.

In this paper, attributes are used as the state transmission mechanism of Jini. HOMI's State Manager operates as a Publisher/Subscriber model like UPnP. This operational method makes a contribution to performance improvements along with Cache Manager. We describe the details of this in Section 3.4. State Manager maintains data consistently relying on Cache Manager at all times and participates in operation of all other modules

3.3 HOMI Language(HOMIL) Analyzer

It is sufficiently feasible for users to construct home network scenario usage by themselves, because their scope is limited to home appliances, the range of usage context is not broad, and previously constructed scenarios are rarely modified in the home network environment unlike ubiquitous environment.

We have designed a script language, HOMIL(HOMI Language), for the convenience of users designing scenarios. We are now working on developing user tools as well. HOMIL is an event-driven processing script language with an ability to design interworking scenarios with a stream of successive events occurred based on a specific event. Scenarios designed using HOMIL is converted into a XML format through the parser and delivered to HOMI.

a) execute Aservice.Method1
b) execute Bservice.Method1 when (Aservice.Method2 == resultC)
c) execute Cservice.method2, Dservice.Method3 when (Bservice.Method2 == resultD)

a) shows pseudo-code using HOMIL. To be specific, a) is a command with only *execute clause*, which executes *Aservice's Method1*. On the other hand, b) is a command with a *when condition clause*, which executes only if the *condition clause* is *true*. The *condition clause* means to execute the *execute clause* if the output is equal to *resultC* after executing *Aservice's Method2*. A command in a *condition clause* can include another commands recursively. c) demonstrates that multiple commands can be used in *the execute clause*. If the *condition clause* is *true*, *Cservice's Method2* and *Dservice's Method3* in the *execute clause* are triggered asynchronously regardless of the time. After HOMIL Analyzer parses a HOMIL program and determines the validity of *when clause*, the *when clause* is registered as an event through Event Manager. Event Manager monitors the occurrence of specific subscribed event and execute services specified in the *execute clause* when the event is occurred and the executed service in turn creates another event for triggering the next service. To summarize, a specific scenario is executed through the successive triggering of these events.

3.4 Cache Manager

One of the problems with the integration of middlewares based on the integrated framework is that the server becomes a bottleneck because all messages must pass through the server. Services exchanging a few message might not be a problem, but services frequently used directly by users or services as scenarios dependent on other devices cause the performance degradation. In this paper, the status of services frequently used by Cache Manager is cached and the unnecessary traffic and subsequently the performance degradation is prevented by Cache Manager, which directly sending the state value to service user instead of querying whenever there is a request from service user. Furthermore, HOMI can overcome a shortcoming of the state advertisement transfer mechanism of UPnP using Cache Manager. In UPnP, in addition to the state variables monitored by Control Point[1], Control Point may generate unnecessary traffic due to status change notification to registered Control Point whenever other variables in the table of state variables are changed. Nevertheless, those data is very useful because the HOMI server need to maintain and cache the status of devices even though such values need currently not to be monitored. Because State Manager maintains the latest state values with the support of State Manager, it responds immediately to user's query.

3.5 Context/Event Manager

Contexts of home network for supporting home automation can be regarded as the conditions which affect the execution of a series of successive scenarios. We have classified contexts into *Time*, *Synchronization*, and *Asynchronization context*. For example, a scenario "The alarm clock turns on at 7:00." is related to the time context.

On the other hand, a scenario "After the alarm turns off, the window is opened." is the synchronization context since the second phrase "the window is opened" is executed after the first phrase "the alarm turns off" is executed and the output is true. From a scenario "After the front door is closed, all the home lights are turned off and the security service is set on.", the scenarios triggered after executing "after the front door is closed" are the synchronization contexts, while the scenario "the lights are turned off" and "the security service is set on" may be executed regardless of the actual ordering of the scenarios. We call these asynchronization contexts. Context Manager and Event Manager always operate in pairs. After classifying scenarios according to the classification, Context Manager sends them to Event Manager. Event Manager manages scenarios for each context with queue structure and triggers an appropriate scenario when the conditions in the context of the scenario are satisfied. The result of the executed scenario changes the status of the device and the table of state variables of State Manager is updated due to the changed status. This change generates an event to trigger the next scenario or is cached by Cache Manager if it is a state variable of frequently modified devices.

4 Scenario Test

A testbed has been constructed in order to show that the proposed scenario-based user-oriented integrated framework is capable of modifying scenarios according to user's demands and the heterogeneous services are interoperable for home automation. First of all, in order to demonstrate the interoperability among heterogeneous middleware services, we developed applications as Jini devices to simulate Jini-Clock which can set the time and turn on and off the clock, and Jini-Light which can remotely turn on and off the light. We also developed an UPnP digital picture frame on the PXA255-based Arm board using UPnP-Light provided by Intel.

Next, in order to demonstrate that services are interoperable using scenarios that user make out for home automation, we construct a scenario: "When Jini-Clock strikes 8:00AM, UPnP-Light and Jini-Light are turned on. When UPnP-Light is on, one of the new stored photos replaces the old one periodically.".

This scenario is represented using HOMIL as shown below.

```
execute UPnPLight.TurnOn(), JiniLight.PowerOn() when
(JiniClock.GetTIme() == 08:00

execute UPnPAngle.ReplaceNewPicture(RANDOM) when
(UpnPLight.GetState() == TRUE)
```

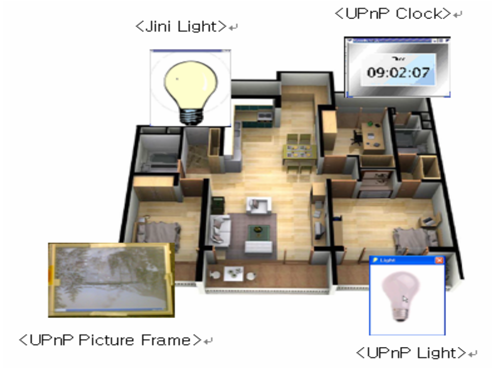


Fig. 3. Testbed for the proposed framework

After constructing heterogeneous applications inside home as shown in Fig 3 to manifest interoperability our experiment tested whether home automation services change according to user's request by testing various scenarios as interworking scenarios are modified. The experiment proved that heterogeneous devices operate successively interacting with each other, following the constructed scenarios. They also operated successfully as we change the ordering of scenarios. Refer to [17] for further details of service interoperation implementation.

5 Conclusions and Future Works

Problems raised by the heterogeneity of home network middlewares must be solved. Also home automation services which are reconstructed using various services through the interoperation of various middlewares upon user demand are very important. In this paper, we have proposed a scenario-based user-oriented integrated architecture in order to meet demand of home network user for various home automation service. HOMI assists users with designing desirable scenarios in flexible ways and receiving seamless services by applying modified scenarios to home automation environment in real-time without installing new applications, updating the server, or rebooting in order to adjust new home network usage scenarios.

In the near future, we will extend contexts to provide support for the execution of home automation services under diverse environments and also HOMI to support the extended contexts. Above all things, among various contexts we will investigate on User contexts and Location contexts which support reloading of user scenarios registered with HOMI server through user authentication and provide services which are most suitable for users by analyzing user's profile and location. Finally, we will also work on developing a smart controller operated by a simple interface based on a WISWIG(what you see is what you get) interface in order for users to design and modify home automation scenarios more conveniently and easily.

References

1. UPnP Forum. http://www.upnp.org.
2. Sun Microsystems. "Jini Architecture Specification". http://www.sun.com/jini/.
3. The Havi Organization "Havi Version 1.1 Specification". http://www.havi.org.
4. Echelon Co., "LonTalk Protocol Specification Version 3.0," 1994.
5. E. Guttman, C. Perkins, J. Veizades and M. Day, *Service Location Protocol*, Version 2 (1990).
6. B Rose, "Home Networks: A Standards Perspective," IEEE Communications Magazine, pp. 78-85, Vol. 39, December 2001.
7. G. O'Driscoll, *The Essential Guide to Home Networking Technologies*, Prentice-Hall, 2001.
8. S. Huhns, Service-Oriented Computing, WILEY,2005 B. Guillaume, R. Kumar, B. Helmut, and S. Thomas, "Methods for bridging a HAVi sub-network and a UPnP subnetwork and device for implementing said methods," Thomson Multimedia, 2002.
9. J. Allard, V. Chinta, S. Gundala, G. Richard III , "Jini Meets UPnP: An Architecture for Jini/UPnP Interoperability," Symposium on Applications and the Internet, pp. 268-275, January 2003.
10. D. Box, "Simple Object Access Protocol 1.1" available at URL http://www.w3.org/TR/SOAP/.
11. E. Tokunaga, H. Ishikawa, M. Kurahashi, Y. Morimoto, and T. Nakajima, "A Framework for Connecting Home Computing Middleware," ICDCSW, pp.765-770, July 2002.
12. OSGI Alliance. http://www.osgi.org/.
13. Kyeong-Deok Moon, Young-Hee Lee, and Young-Sung Son, and Chae-Kyu Kim, "Universal Home Network Middleware Guaranteeing Seamless Interoperability among the Heterogeneous Home Network Middleware," IEEE Transactions on Consumer Electronics, Vol. 49, August 2003.
14. R. Ponnekanti and A. Fox. "Application Service Interoperation without Standardized Service Interfaces,“ Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, pp. 30-40, March 2003.
15. KISDI, *An Analysis, of and Prospect of Home Networking Market*, December 2003 (In Korean).
16. Min Chan Kim, *A Study on Next Generation Home Network Middleware and Security*, Internal Report #5, August, 2005 (in Korean).

Session Key Agreement Protocol for End-to-End Security in MANET*

Jeong-Mi Lim[1] and Chang-Seop Park[2]

[1] Department of Computer Science, Dankook University,
Anseo-dong, Cheonan, Chungnam, 330-714, Korea
redpig3@dankook.ac.kr
[2]Department of Computer Science, Dankook University,
Anseo-dong, Cheonan, Chungnam, 330-714, Korea
csp0@dankook.ac.kr

Abstract. Mobile ad hoc network (MANET) is an infrastructure-less network, consisting of wireless nodes without access points or base stations. Since mobile nodes in MANET move very easily and freely, MANET is appropriate for ubiquitous environment. But, from a security viewpoint, MANET is a very weak network since various security attacks against it such as eavesdropping or DoS (Denial-of-Service) attack can be more easily performed than against the wired network. In this paper, we design a key agreement protocol for end-to-end security between source node and destination node without any security infrastructure. Diffie-Hellman key agreement mechanism is combined with a concept of CGA (Cryptographically Generated Address) mechanism to provide source authentication service. Based on the IPv6's IP auto-configuration, how to generate IPv6 address from the Diffie-Hellman key pair is explained, and a mechanism to generate session key for both authenticating nodes and protecting messages exchanged between them is suggested. We also evaluate the performance of our mechanism using NS2 (Network Simulator).

1 Introduction

Ubiquitous environment means “whenever”, “wherever”, namely, regardless of place and network environment, somebody and something can join the network. Recently, research on ad hoc network which is one of ubiquitous network fields goes on. Ad hoc network is divided into two parts, mobile ad hoc network (MANET) and sensor network. Ad hoc network consists of only wireless nodes without any infrastructure. Since it consists of only wireless nodes, nodes move dynamically, easily, freely. So MANET is useful application for natural disaster, emergency services in military, conference, meeting room, and home network, etc.

When a source node wants to send messages to a destination node which is more than one-hop away from it, intermediate nodes between them act as routers in order to relay the messages from the source node to the destination node. From a security viewpoint, MANET is a very weak network since various security attacks against it

* This work was supported (in part) by the Ministry of Information & Communications, Korea, under the Information Technology Research Center (ITRC) Support Program.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 679 – 686, 2006.

such as eavesdropping or DoS (Denial-of-Service) attack can be more easily performed than against the wired network.

However, security measures used for protecting wired network cannot be directly employed for MANET. First, nodes in MANET usually have limited computing resources in terms of battery, memory and CPU, which means computationally-expensive operations such as asymmetric cryptographic operation cannot be performed. Second, nodes in MANET usually move very dynamically, and furthermore some nodes join and leave from MANET frequently, so that the network topology is also changed very rapidly. Therefore, distributed security measures should be considered without any security infrastructure such as centralized CA (Certificate Authority) or TTP (Trusted Third Party), for the purpose of protecting MANET.

Security requirements in MANET contain not only phase of discovering routing path but also data transmission. In this paper, we assume a secure routing protocol has already existed, and propose a session key agreement between source node and destination node, in order to secure data transmission.

This paper is organized as follows. In Section 2, we present IP address auto-configuration concept of CGA (Cryptographically Generated Address) [1] in ad hoc network, and other mechanisms of key management and authentication. In Section 3, we propose a session key agreement protocol for end to end using Diffie-Hellman key agreement mechanism based on the IPv6's IP auto-configuration. In Section 4, we evaluate the performance of our mechanism using NS2. In Section 5, concluding remarks are given.

2 Related Works

2.1 IPv6 Address Auto-configuration in Ad Hoc

IPv6's IP address allocation mechanism [4] can be stateful or stateless. In stateful method, DHCP allocates IP address to node. On the other hand, in stateless method, each node generates IP address by itself. The latter is suitable for MANET environment. Concept of the CGA is another method to generate IP address using node's public key and hash function, for example, MD5 or SHA-1.

Node which wants to join in MANET, generates public key and private key pair (*PK, SK*). IPv6 address using a concept of CGA [1], consists of 64bits prefix and *Hash*(*PK, m*)., as in Fig. 1, where *HASH*( ) is a one-way hash function and *m* is a random number in order to avoid duplication.

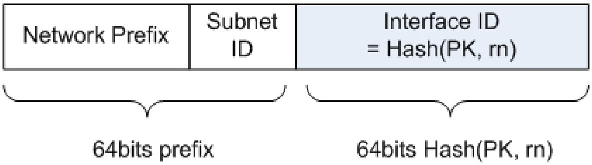


Fig. 1. CGA-based IPv6 address

To join MANET, node has to process DAD (Duplicate Address Detection) [3]. NS (Neighbor Solicitation) and NA (Neighbor Advertisement) messages in IPv6 correspond to AREQ (Address Request) and AREP (Address Reply) messages in MANET. Namely, to process DAD, each node broadcasts AREQ message, and if another node in MANET has the same IP address, then the node unicasts AREP message to the requesting node.

2.2 Key Management and Authentication

In MANET environment, we can't use administrative CA (Certificate Authority), TTP (Trusted Third Party), KDC (Key Distribution Center), because network topology is dynamically changed. So, recently, researches on distributed approach without centralized security authority have been proposed, such as threshold secret sharing, self securing, statistically unique and cryptographically verifiable (SUCV) identifiers.

In threshold secret sharing [7], CA capacity is distributed. If a single CA is responsible for entire nodes in MANET, then the CA node becomes a target of security attack and is also a source of bottleneck. In $(n, t+1)$ threshold secret sharing, there are n servers (server$_1$, server$_2$, ..., server$_n$.). Each server$_i$ has a key pair, private key k$_i$ and public key K$_i$. Server$_i$ generates a partial signature $PS(m, s_i)$. If more than $t+1$ partial signatures are combined, then we can authenticate the node..

Self-securing [8] uses PGP which follows "web of trust" authentication model. Each node generates its own public key and private key. If there are intermediate nodes A, B, C between source node S and destination node D, then, the node S authenticates the node A, the node A authenticates the node B, the node B authenticates the node C, and the node C authenticates the node D. As a result, the node S authenticates node D.

Statistically unique and cryptographically verifiable identifiers [6][9] use a binding node's IP address and public key. This mechanism is processed in security bootstrapping phase.

3 Proposed Protocol

Communication in MANET consists of two phases, route discovery phase and data transmission phase. It is important to secure data transmission as well as to secure route discovery. In this section, we assume a secure route discovery mechanism has existed, and propose a secure session key agreement for end-to-end nodes, in order to secure data transmission.

We use a Diffie-Hellman key agreement mechanism, and a public key and a MAC (Message Authentication Code) value are inserted in RREQ (Route REQuest) / RREP (Route REPly) message. In Fig. 2, the source node sends $<RREQ, g^s>$, where g^s is a source node's public key and g^d is a destination node's public key. The destination node computes $(g^s)^d$ with g^s and destination node's private key d. In order for key refresh, this mechanism uses bid, and the hash function is used to maintain a small key size. The session key is computed as $Hash((g^s)^d \mid\mid bid)$. *bid* is a broadcast id in RREQ, so that the session key is changed at every session. The destination node unicasts RREP and $MAC_{sessionkey}(RREP\mid\mid g^d)$ to the source node, where, $MAC_{session\text{-}key}(RREP\mid\mid g^d)$ represents MAC (Message Authentication Code) value which is computed with session key to protect RREP message and public key g^d.

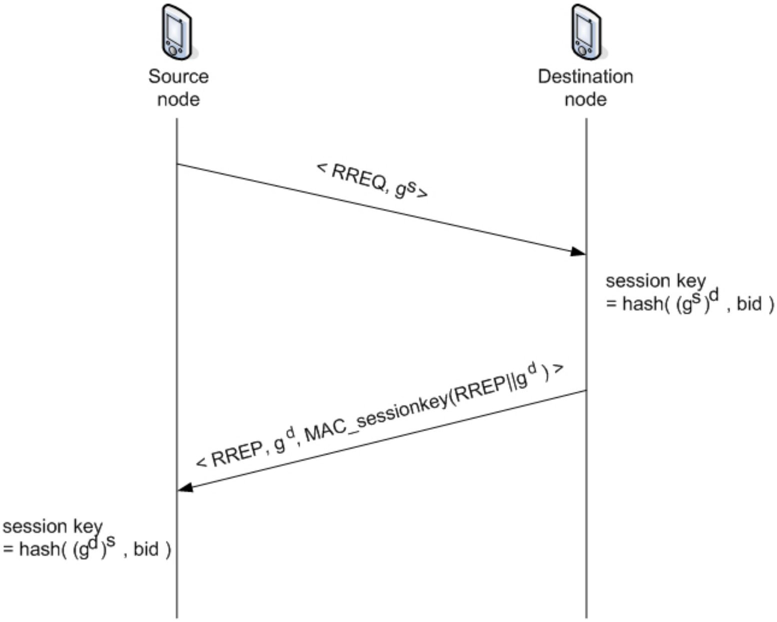

Source node
Destination node
< RREQ, gs >
session key = hash((gs)d , bid)
< RREP, gd, MAC_sessionkey(RREP||gd) >
session key = hash((gd)s , bid)

Fig. 2. Session key establishment

The key agreement method based on the plain Diffie-Hellman method can be a target of the-man-in-the-middle attack. But, in this paper, the IP address is bind with the public key. Like in figure 3, suppose a malicious intermediate node forges $< RREQ,\ g^s >$ into $< RREQ,\ g^m >$ and sends the forged message to the destination node.

The destination node's verification process is as follows.

1. Compute $Hash(g^m)$ with g^m.
2. Compare $Hash(g^m)$ to 64 bits suffix of source node IP address in RREQ.

If two values are matched, then intermediate node is valid, but, here 64 bits suffix of source node IP address in RREQ is $Hash(g^s)$ since source node's public key is g^s. So, intermediate node is malicious node and the man-in-the-middle attack can be detected.

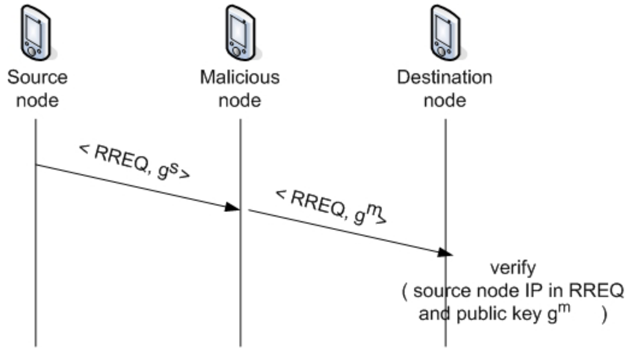


Fig. 3. Detection of man-in-the-middle attack

4 Simulation and Analysis

Proposed protocol in section 3 is simulated with NS2 network simulator with CMU Monarch extension [2]. Diffie-Hellman and MAC value generation is implemented by openssl [5]. Simulation environment is shown in Table 1. We have 10 source nodes among a total of 50 nodes, and mobility model is random way point model. Pause time means mobility. If pause time is 0, then mobility goes on. If pause time is increased, then mobility is slowly.

Table 1. Environment for simulation

Nodes	50
Number of sources	10
Scene	1000m X 1000m
Simulation time	200sec
Maximum velocity	20m/sec
Transmission range	250m
Mobility model	Random way point
Pause Time	0, 20, 40, 60, 80, 100
Data	512 bytes CBR data
Traffic	4 pkts/sec

We evaluate routing overhead, normalized routing load, and end to end delay per pause time. Simulation result and analysis is as follows. In Fig. 4, AODV is basic AODV protocol, and make session key is the proposed AODV with session key agreement.

4.1 Routing Overhead

Routing overhead means the number of control messages. If the pause time is decreased, then mobility is fast, and network topology is dynamically changed. So, the number of control messages is increased. If pause time is increased, then mobility is slowly. So, the number of control messages is decreased. Since the public key and MAC value is sent with RREQ/RREP message, the number of control messages is not increased. In Fig. 4, the proposed protocol's simulation result is similar to basic AODV protocol.

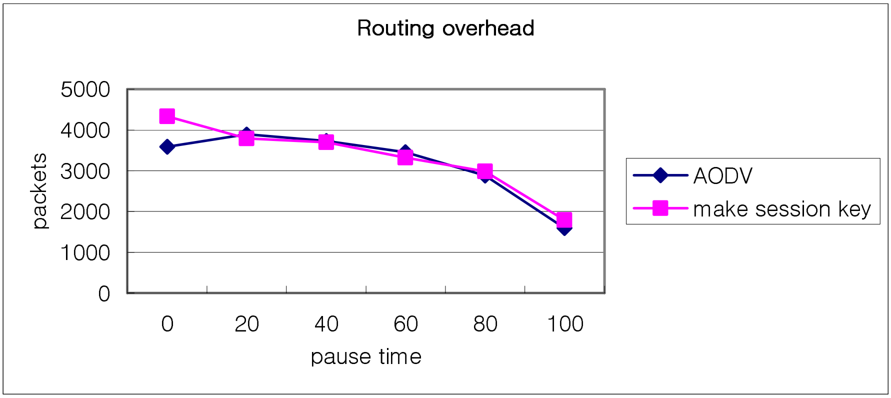


Fig. 4. Routing overhead

4.2 Normalized Routing Load

Normalized routing load means a control message ratio per received packet. In Fig. 5, because of same reasons in routing overhead, proposed protocol's simulation result is similar to basic AODV.

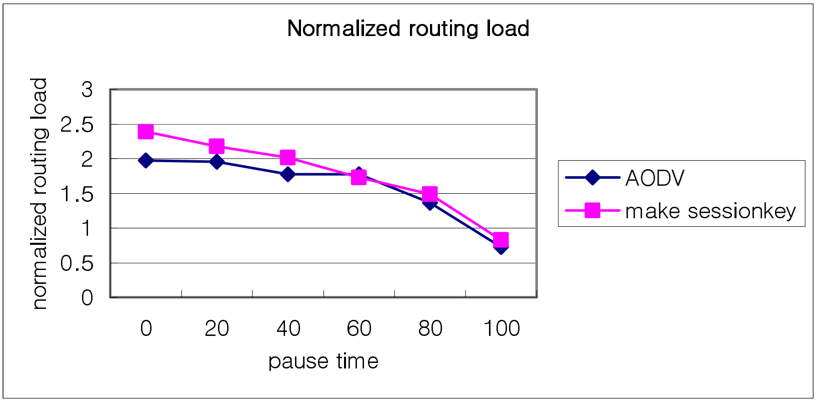


Fig. 5. Normalized routing load

4.3 End-to-End Delay

End-to-end delay means the duration between sending start time and receiving end time per a received packet in end-to-end. Delay for both computing $(g^s)^d$ or $(g^d)^s$ and verifying $MAC_{sessionkey}(RREP||g^d)$ are added.

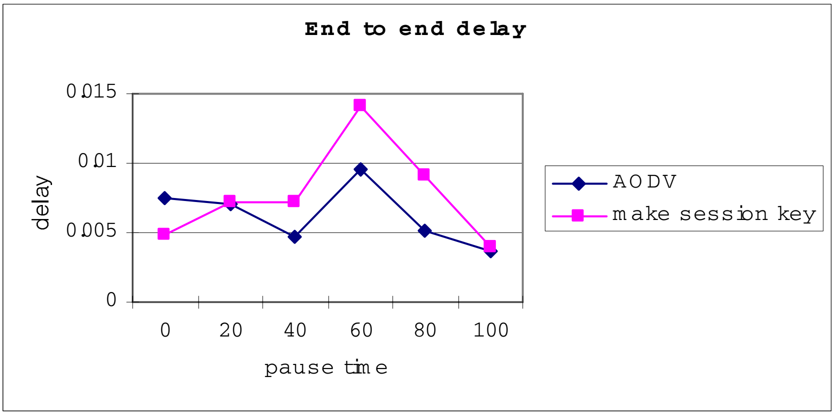


Fig. 6. End to end delay

In Fig. 6, proposed protocol's end-to-end delay is longer than that of basic AODV protocol.

In the above three simulation results, it can be known that the proposed protocol has a similar routing overhead to basic AODV, and little delay. The reason is that both public key and MAC value are sent with RREQ/RREP message, based on basic AODV. Public key size is 1024 bits and MAC value size is 128bit or 160 bits. In this experiment, MD5 is used, so MAC value size is 128bits. Also, delay is little, because of using MAC instead of signature.

5 Conclusions

In this paper, we proposed a session key agreement protocol for end-to- end security in MANET, which is one of the ubiquitous environments. To authenticate nodes which are notebooks or PDAs or other mobile devices, and to establish session key between source node and destination node, we use both Diffie-Hellman Key agreement and a concept of CGA. Since MANET does not employ the administrative authorities such as CA, TTP, or KDC, the IP address auto-configuration based on CGA is suitable for MANET. The proposed protocol protects from man-in-the-middle attack, and does not require additional control message.

References

1. Aura T.: Cryptographically Generated Address (CGA), RFC 3972, Work in Process, March (2005)
2. CMU Monarch Group.: CMU Monarch Extensions to the NS Simulator, http://mnarch.cs.cmu.edu/cmu-ns.html, August (2002)

3. C.E.Perkins, J.T.Malinen, R. Wakikawa, E.M. Belding Royer, and Y.Sun.: IP Address Autoconfiguration for Ad Hoc Networks, draft-ietf-manet-autoconf-01.txt, IETF MANET Working Group, (2001).
4. S. Thomson, T. Narten.: IPv6 Stateless Address Autoconfiguration, Network Working Group RFC 2462, December (1998).
5. John Viega, Matt Messier, Pravir Chandra.: Network Security with OpenSSL, O'REILLY.
6. Yu-Chee Tseng, Jehn-Ruey Jiang, Jih-Hsin Lee.: Secure Bootstrapping and Routing in an IPv6-Based Ad Hoc Network, ICPP Workshops (2003).
7. L. Zhou and Z. Haas.: Securing Ad Hoc Networks, IEEE Network, vol. 13, no.6, pp.24~30, Now./Dec. (1999).
8. Haiyun Luo, Petros Zerfos, J. Kong, Songwo Lu, Lixia Zhang.: Self-Securing Ad Hoc Wireless Networks, Seventh IEEE Symposium on Computers and Communications (ISCC '02), (2002).
9. G. Montenegro and C. Castelluccia.: Statistically Unique and Cryptographically Verifiable (SUCV) Identifiers and Address, Proc. Ninth Ann. Network and Distributed System Security Symp. (NDSS), (2002).

Process-Oriented DFM System for Ubiquitous Devices

Yongsik Kim[1], Taesoo Lim[2], Dongsoo Kim[3,*], Cheol Jung[1], and Honggee Jin[1]

[1] Production Engineering Research Institute, LG Electronics Inc., 19-1 Cheongho-ri,
Jinwuy-myun, Pyungtaek-si, Kyunggi-do, Korea
{kimyongsik, jungc, jinhg}@lge.com
[2] Department of Computer Science, Sungkyul University, Anyang-8 Dong, Manan-Gu,
Anyang-city, Kyunggi-Do, Korea
tshou@sungkyul.edu
[3] School of Industrial and Information System Engineering, Soongsil University,
1-1 Sangdo-dong, Dongjak-Gu, Seoul, Korea
Tel.: +82-2-820-0688; Fax.: +82-2-825-1094
dskim@ssu.ac.kr

Abstract. As advanced multimedia and communication technologies such as DMB (Digital Multimedia Broadcasting) and bluetooth technologies are commercialized, the life cycle of ubiquitous devices is getting shorter and shorter. In this market environment, reducing the product development lead time with high quality is one of the most important key success factors to accomplish both product design and technology leadership. Product manufacturability evaluation activities in earlier design stage can contribute to the reduction of redundant and repetitive works, specifically, the number of design changes during mass production phase. To facilitate the manufacturability evaluation activities, interdepartmental collaboration processes and channels for sharing design knowledge need to be established. In this paper, we suggest a collaboration process for manufacturability evaluation using checklist devised for ubiquitous devices. Also, we have implemented a web-based collaborative DFM (Design for Manufacturability) system to share both design knowledge and evaluation results in real-time.

1 Introduction

Recently, as multimedia and mobile communication technologies are evolving rapidly, the life cycle of high-tech products is getting shorter and shorter. Therefore, it is very important for companies to establish product design and technology leadership and to lead the market by releasing new products in the market at right time. Not only the market leadership but also mass production capability of high quality products is very significant factor to achieve competitiveness of products and the company. To cope with such requirements, it is very important to minimize manufacturing problems by supporting the design phase effectively, which results in minimizing design changes in the mass production phase.

[*] Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 687 – 696, 2006.

Eighty percent of manufacturing costs and time are determined in the product design stage. Design changes due to the problems occurring in mass production process require much cost and time [6]. In product development stage, it is required to design products considering not only performance and reliability of products but also production environments. Concurrent engineering methodology for product development processes is one of the methods to accomplish it.

DFM (Design for Manufacturability) is a concurrent engineering methodology in the perspective of manufacturability. It helps to identify manufacturing problems in advance and solve the problems, which enables shortening of product development lead time and improving mass production quality [1]. Applying the DFM methodology to product development processes requires inter-departmental collaborative processes for verification and improvement of product manufacturability, and common channels for sharing design knowledge such as circuit drawings, mechanical drawings, and softwares.

The purpose of this paper is to establish collaborative process for the manufacturability verification of ubiquitous devices using DFM checklist and develop a web-based DFM system for sharing product development knowledge and verification results. With the result of this work, it is possible to apply DFM methodology to the production ubiquitous devices and rapidly verify the quality of mass production.

The rest of the paper is organized as follows. Section 2 describes related researches and approaches focusing on DFM metric. Section 3 presents the composition and contents of the DFM checklist. Section 4 explains DFM the collaborative process using the checklist. Section 5 summarizes the development of the web-based DFM system. Finally, Section 6 concludes this paper.

2 DFM Metrics

DFM is a methodology for the verification of manufacturability. Several tools and techniques are widely applied such as CE (Cost Engineering), QFD (Quality Function Deployment) and GT (Group Technology). Also, DFM guidelines and rules have been published and widely used by many companies [6].

Table 1 illustrates major DFM metrics related with DFM implementation rules suggested by Jami J. Shat, et al. These methodologies assign priority calculated with weight and cost in part selection and design determination considering manufacturing conditions.

We present a DFM checklist for manufacturing ubiquitous devices using the considerations suggested in Table 1 and reinforcing quality problems and process constraints for embodying verification of manufacturability knowledge. Using the checklist we have developed a web-based collaborative information system for the verification of mass production manufacturability. The manufacturability knowledge does not exist as a fixed form and should be enhanced continuously through information sharing among disciplines such as design, manufacturing, marketing department. Especially, for the development of ubiquitous devices with speedy technology innovation, continuous inter-departmental collaboration processes rather than fixed knowledge is highly required.

Table 1. Major DFM metric

DFM Metric	Descriptions
Qualitative scores based on good practice rules	It quantitatively evaluates how well practice rules with weighted priority according to importance are reflected on the product design. Practice rules consist of check items needed to design product for mass production. Specifically, it involves selection criteria for parts and materials considering process constraints, assemblability, and mechanical processes.
Direct cost estimates and time based manufacturability rating	It calculates production cost per product based on the average cost of the parts and materials in the design phase. Standardization and sharing of parts are utilized to reduce costs. When time-to-market or lead time is more important factor to be considered, the manufacturability is evaluated based on required time.
Design tolerance to process capability ratios	It evaluates production manufacturability based on throughput yield considering process capability index (Cpk) or statistical variation of processes. It adjusts design parameters within the boundary of product specification.
DFM based on Taguchi loss function	It utilizes DOE for setting up optimized design parameters. Tolerance design is performed to minimize the loss of customers (or manufacturers) due to the change of design parameters.

There have been several cases applying collaborative DFM methodology for the development of commercialized products, which includes automobile industry case suggested by Hiroshi Onisuka, et. al. They classified activities needed for manufacturing Nissan Maxima engines into ordinary activities and difficult activities. They applied discriminated improvement tools and techniques for each activity and proposed an approach for minimizing total manufacturing time and costs. With UNITEC (Unit Trial Production Technical Meeting) devised for the collaboration between design and manufacturing departments, they identified target items for reducing time and costs for each prototype and continuously reflected them on the product design, which resulted in achieving high manufacturability throughout the development process.

3 The DFM Checklist

As described in Section 2, existing DFM researches are focused on reducing time and costs for mass production using part sharing and assemblability improvement methods. However, to minimize the design changes in the production phase, both quality problems which might occur under mass production and constraints of each manufacturing process must be considered in the design phase. The DFM checklist proposed in this work is composed of check items for preventing re-occurrence of mass production problems already experienced and verification items for satisfying manufacturing process constraints, which should be reflected on the product design. The DFM checklist is the metrics for evaluating the production quality of the design and satisfiability of process constraints prior to finalizing the product design.

As is shown in Fig. 1, manufacturing processes of mobile devices, which are predominant terminal devices for ubiquitous computing, consist of SMT (Surface Mount Technology) process, sub assembly process, total assembly process and inspection process. SMT process is the process for making PCB (Printed Circuit Board) composed of major chipsets and circuits of mobile devices and storing up multiple PCBs if necessary. Recently, as a variety of multimedia technologies are applied for the production of mobile devices, the complexity of SMT process is increasing.

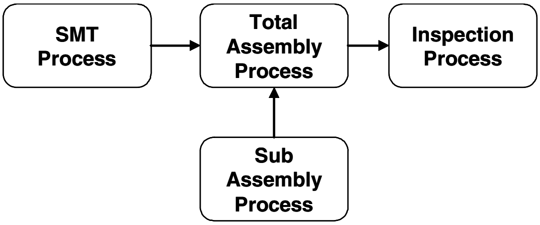


Fig. 1. Manufacturing processes of mobile devices

Sub assembly process is for assembling mechanical parts, and total assembly process is for assembling the sub parts made by sub assembly process and the PCB made by SMT process. Inspection process is composed of software downloading and overall software testing. Table 2 illustrates design technology areas for the manufacturing processes of mobile devices

Table 2. Manufacturing processes and related design areas

Manufacturing process	Design areas
Total assembly, sub assembly	Mechanics and hardware
SMT	Electronic packaging
Inspection	Software

The DFM checklist is a practical guideline developed through analyzing causes of production problems which occurred in actual production processes. It is composed of check items in the area of mechanics, hardware, electronic packaging, and software. Check items of each technology area are classified by its unique classification scheme. For example, the classification structure of checklist of mechanics is illustrated in Fig. 2.

Check items of each technology area can be modified through continuous monitoring of mass production problems or changes in process constraints. Examples of check items in mechanics and hardware area are shown in Table 3. Check items for production manufacturability consist of issue list and response plan identified by analyzing problems which occurred in the prototype or mass production stage.

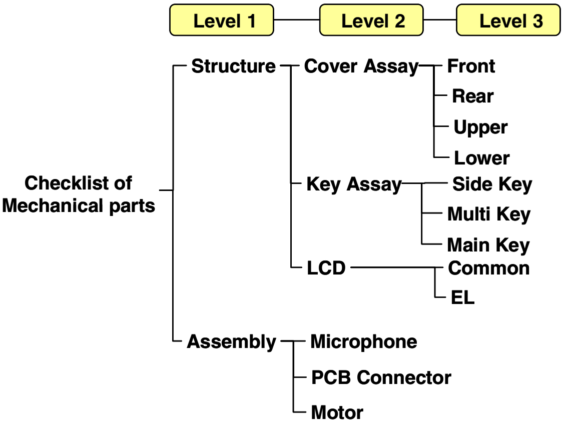
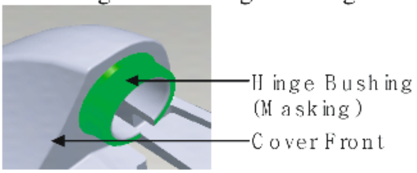
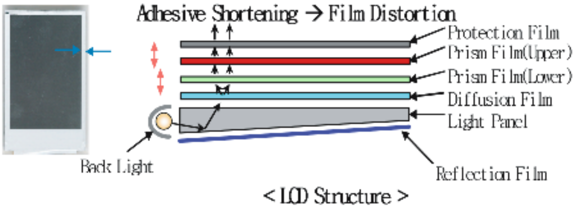


Fig. 2. The classification structure of the checklist of mechanical part

Table 3. Manufacturability evaluation criteria for mechanics and hardware

Area	Check item	Evaluation criteria
Mechanics	Verification of size of hinge holder insertion part	- Size checking of hinge holder and lower hinge holder - Masking of front hinge bushing 
H/W	Optical sheet of LCD is distorted in drop testing	- making paste surface of each film more wide than XX mm - extending the protrusion for gripping prism sheet  Adhesive Shortening → Film Distortion Protection Film Prism Film(Upper) Prism Film(Lower) Diffusion Film Light Panel Back Light Reflection Film < LCD Structure >

Weighted priorities are assigned to each check items and the result of evaluation is 'OK', or 'NG'. The DFM metric is defined as the level of completeness in the manufacturability. The value of completeness of mass production can be calculated using the following formula.

$$C = \frac{\sum_{i=1}^{n} w_i r_i}{\sum_{i=1}^{n} w_i} \tag{1}$$

In the formula, w_i is weighted priority assigned to i-th check item. And, r_i is the result of i-th check item and it is an encoded quantitative value transformed from the qualitative test result.

4 The Collaborative DFM Process

To properly evaluate manufacturability in the earlier design stage, each step in the product development process involving product planning, selection of parts and detail design should be performed according to the manufacturability perspectives. Therefore, concurrent and collaborative design process can play an important role in guaranteeing the manufacturability in the earlier design stage.

This paper proposes a concurrent and collaborative evaluation process for the manufacturability and quality of the products in the earlier design stage. The evaluation process is made up of verification of the problem, quantification of the problem, and design optimization to minimize the problem, and is repeated *n*-times until the quality reaches the target level (Fig. 3).

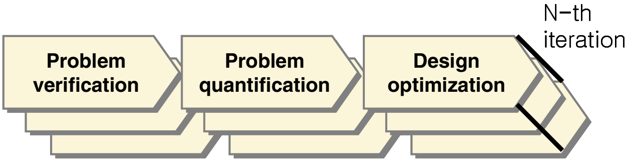


Fig. 3. Manufacturability evaluation process

Mobile device development process consists of a series of PCP(Product Concept Planning), PP(Project Planning), DV(Design Verification) and PQ(Product Qualification) stages. Among the stages, collaborative DFM process starts with FMEA(Failure Mode Effect Analysis) in PP stage, and ends with manufacturability review in PQ stage. Fig. 4 illustrates the relationship between product development process and collaborative DFM process.

Specific works performed in each collaborative step are described as follows.

- Step 1. Process FMEA in PP stage

PP step is the stage to evaluate the possibility of realization of product concept before actual design, in which both design FMEA and the process FMEA are performed. Not only designers but also production engineers participate in process FMEA to review the previous problems and apply its measures into the DV design stage. The FMEA utilizes the DFM checklist of the similar product models, and checks the design changes due to the process constraints and quality problem in detail.

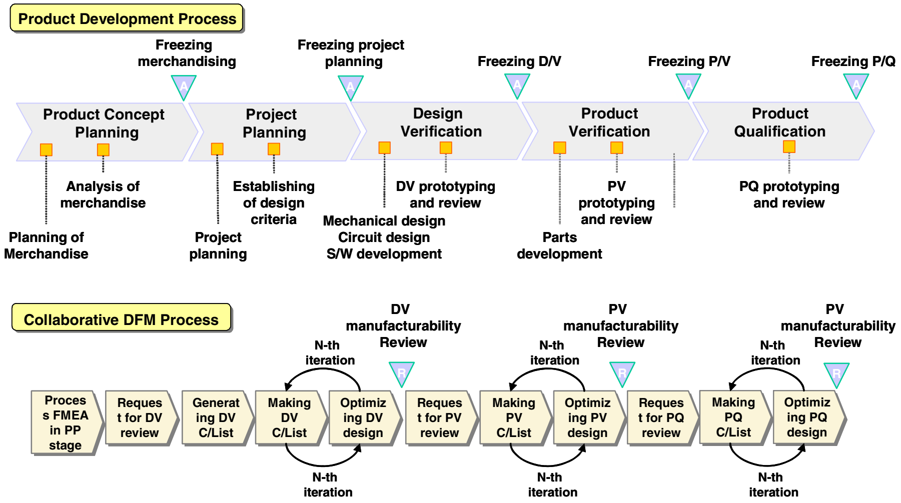


Fig. 4. Product development process and collaborative DFM process

- Step 2. Requesting manufacturability evaluation in each DV, PV, and PQ stage

Designers make a request for manufacturability evaluation to pilot engineers through the existing PDM (Product Data Management) system. The request is independently performed in each technology area like mechanics, H/W, S/W, and PCB. The reason to use PDM system is to easily pass the related design knowledge such as mechanic and circuit drawings. PDM is used as a channel for shareing design knowledge among designers, pilot and manufacturing engineers. The system manages the design knowledge according to the item level like PCB assembly, sub assembly, S/W and parts as is shown in Fig. 5. Both pilot and manufacturing engineers evaluate the manufacturability using prototype and CAD drawings.

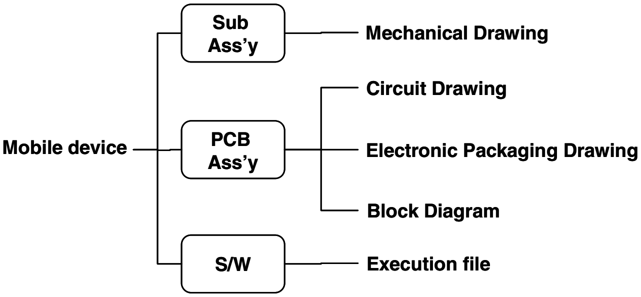


Fig. 5. Item descriptions in PDM system

- **Step 3**. Generating the Checklist in DV stage

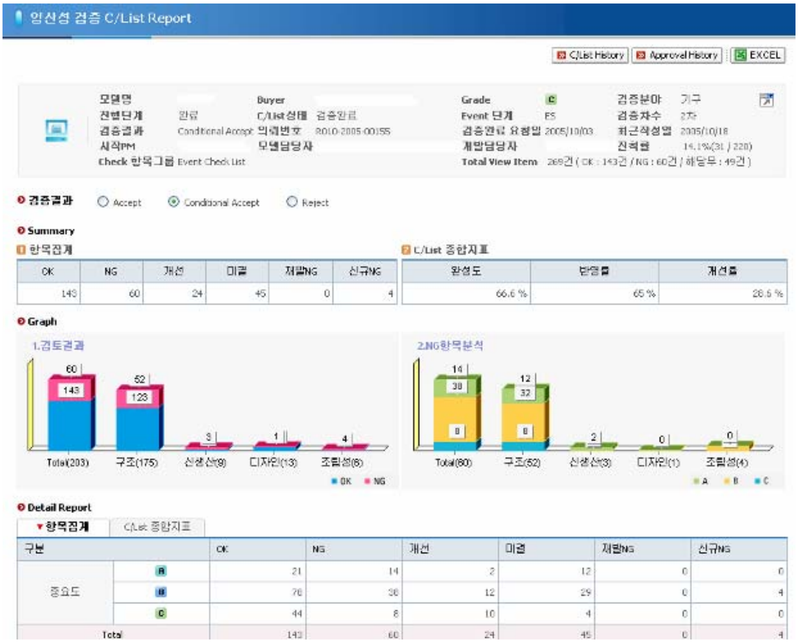
The pilot engineer examines the evaluation request, makes go-or-no decision, and generates the checklist. Evaluation items in the checklist are dynamically organized considering functional characteristics of the model being checked. For example, if the model supports DMB function, evaluation items related to the DMB function are dynamically selected to make up the list.

- **Step 4.** Making up the checklist in DV, PV, and PQ stage

Both pilot and manufacturing engineers investigate each prototype using design knowledge such as circuit drawings, mechanical drawings, process constraints, and softwares, and they feedback the evaluation results to the checklist. The results consist of detail phenomena and its causes made by referencing to design guide as well as review criteria and specification described in the checklist. If the result is 'NG', the engineer makes a request for design improvements and asks for the designer to input the measures for the problem. All the inputs to the checklist are version-controlled.

- **Step 5.** Design optimization in DV, PV, PQ stages

Designer investigates whether the improvement request is possible to reflect it into the design, and inputs the examination result for the improvement plan in the check list. Both pilot and manufacturing engineer verify the plan, and the actual design change using design knowledge.



양산성 검증 C/List Report

모델명		Buyer		Grade	C	검증분야	기구
진행단계	완료	C/List상태	검증완료	Event 단계	ES	검증차수	2차
검증결과	Conditional Accept	의뢰번호	R010-2005-00155	검증완료 요청일	2005/10/03	최근작성일	2005/10/18
시작PM		모델담당자		개발담당자		진척률	14.1%(31 / 220)
Check 항목그룹	Event Check List			Total View Item	269건 (OK : 143건 / NG : 60건 / 해당무 : 49건)		

검증결과 ○ Accept ● Conditional Accept ○ Reject

Summary

항목집계

OK	NG	개선	미결	재발NG	신규NG
143	60	24	45	0	4

C/List 종합지표

완성도	반영률	개선률
66.6 %	65 %	28.5 %

Graph

Detail Report

▾항목집계 C/List 종합지표

구분		OK	NG	개선	미결	재발NG	신규NG
중요도	A	21	14	2	12	0	0
	B	76	38	12	29	0	4
	C	44	8	10	4	0	0
Total		143	60	24	45	0	4

Fig. 6 Analysis of manufacturability evaluation results

- **Step6.** Manufacturability review in DV, PV, and PQ stage

Manufacturability review is a procedure for the designer, pilot and manufacturing engineer to review the checklist results, and to establish future plan for the defects in current product development process (Fig. 6). If there are so many design changes, prototype is re-fabricated and the review is repeated.

5 Web-Based DFM System

We have developed a web-based system for collaborative DFM process, and the system has been operated as an application module in overall e-R&D enterprise portal of a company. We used SSO (Single Sign On) policy in the portal as basic security mechanism. The system has a typical 3-tier architecture based on J2EE.

Since addition and modification of checklist items should not influence the product development process, we implemented rapid reactive system to adjust to the item changes due to the introduction of new operation or the modification of existing operation. Fig. 7 illustrates logical view of the system architecture and information flow.

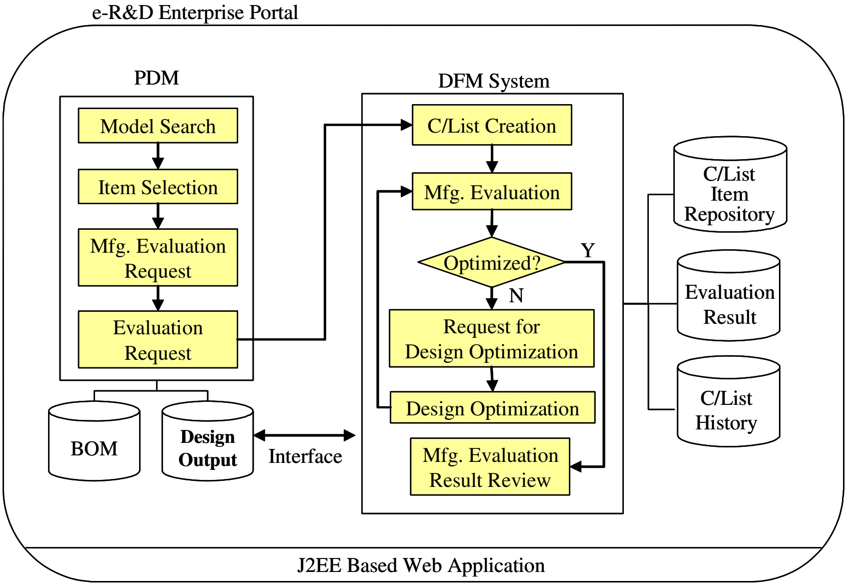


Fig. 7. Logical view of the system architecture of the DFM system

6 Conclusion

The collaborative DFM process for the design of ubiquitous devices considering mass production manufacturability and web-based information system supporting the collaborative process are proposed in this work. The collaborative DFM process is adopted to utilize know-how of design, production engineering and manufacturing in the perspective of mass production in the product design phase. The benefits of the collaborative process can be summarized as follows. First, by incorporating mass production manufacturability in design phase, costs of product failure can be reduced. Second, design changes due to the process constraints in mass production stage can be minimized. Third, the verification level of mass production manufacturability can be enhanced enterprise-widely using the standard checklist and design guideline. Finally, the flow of design knowledge is optimized through the integration of the PDM and DFM system.

A company using the collaborative DFM system proposed in this paper has accomplished 80 % of completeness of P/V stage manufacturability in manufacturing ubiquitous devices and failure rate of the first mass production has been reduced significantly. Especially, manufacturing process constraints incorporated by new automation production line can be analyzed thoroughly in advance and identified as check items prior to actual production, which enables reflecting possible problems due to process constraints on the product design rapidly.

Check items in the checklist should be organized properly to increase the possibility of prior identification of mass production problems by applying the DFM process. The checklist explained in this work is developed by analyzing problems in the first mass production stage. However, to increase the reliability of the checklist it is required to extend the DFM system by integrating MES (Manufacturing Execution System) and the DFM system and managing process troubles from the product design phase to the mass production phase.

References

1. Gupta, S.K., Regli, W.C., Das, D., Nau, D.S., Automated Manufacturability Analysis: A Survey, Research in Engineering Design, l9 (1997), 168-190
2. Onitsuka, H., Eguchi K., Miura, N., Matsumura, H., Productivity evaluation method and design for manufacturability, JSAE Review, 16 (1995), 375–381
3. Prasad, B., Survey of life-cycle measures and metrics for concurrent product, Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 14 (2000), 163–176
4. Ramaswamy, S., A Survey of DFM Method, available at
http://asudesign.eas.asu.edu/ education/MAE540/sanjayweb.htm
5. Shah, J.J., Wright, P.K., Developing Theoretical Foundations of DFM, Proceedings of DETC 2000: Design For Manufacturing Conference September 10-14 (2000), Baltimore, MD, USA
6. van Vliet, J.W., van Luttervelt, C.A., State-of-The-Art Report on Design for Manufacturing, Proceedings of the 1999 ASME Design Engineering Technical Conferences September 12-15 (1999), Las Vegas, Nevada, USA
7. Zhao, Z., Shah, J., A Normative DFM framework based on benefit-cost analysis, Proceedings of DETC 2002: ASME 2002 Design Engineering Technical Conference And Computers and Information in Engineering Conference, September 29-October 2 (2002), Montreal, Canada,

A Study on the Application of BPM Systems for Implementation of RosettaNet Based e-Logistics

Yong Gu Ji[1], Chiwoo Park[2], and Minsoo Kim[3,*]

[1] Department of Information & Industrial Engineering, Yonsei University,
134 Sinchon-Dong, Seodaemun-Gu, Seoul, Korea
yongguji@yonsei.ac.kr
[2] TI Consulting Service Line, Deloitte Consulting Korea,
19th floor of Seoul Finance Center Building, Taepyong-Ro, Jung-Gu, Seoul, Korea
chiwoo.park@gmail.com
[3] Department of Systems Management and Engineering, Pukyong National University,
San 100, Yongdang-Dong, Nam-Gu, Busan, Korea
Tel.: +82-51-620-1556 ; Fax: +82-51-620-1546
minsky@pknu.ac.kr

Abstract. With the progress of globalization, supply chains of enterprises have expanded to cover the whole world, and in such environment, enterprises have placed various efforts to enhance the efficiency of their supply chains. For efficient management of supply chains, a number of enterprises have implemented VMI processes and have started to employ TPL as a way of strategic outsourcing. There also have been recent efforts to connect VMI processes with TPL via IT technologies to enhance competitiveness. In such e-Logistics programs, international e-commerce standards such as RosettaNet can be used as a means to intensify the control of logistics and inventory information, but still has limitations in that causal relationship between PIPs are not fully described to execute VMI processes with TPL. This study intends to overcome such limitations by implementing a content based document routing function that connect the RosettaNet B2B system to the BPM system. Furthermore, to monitor the PIP instance's causal relationship, a multi-PIP monitoring system has been developed, which in turn will facilitate the management and control of higher level BPM processes. The results of this study are already being applied to e-Logistics programs of a Korean company, and runs successfully in production mode.

1 Introduction

To survive in the world of intensified competition, enterprises have struggled to maintain their competitiveness and strength while outsourcing minor business functions. With the globalization of the enterprise environment, outsourcing candidates have expanded to include foreign companies, and the supply chains of individual enterprises have become more complex. As a result, enterprises are placing considerable amount of efforts to manage their supply chains with higher efficiency and to construct an optimized logistics system [1, 2, 3].

[*] Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 697 – 706, 2006.

Of these various efforts, e-Logistics project, which consists of TPL (Third Part Logistics) and VMI (Vendor Managed Inventory) as its core process, has attracted much interest due to its high performance result. TPL allows an enterprise to minimize its efforts in its intricate logistic systems while achieving higher logistics performance. In cases of multi-leg supply chains, VMI is an approach that enables enterprises to reduce their supply chain management cost while minimizing safety stock costs for in-between enterprises of the chain [4, 5].

The efficacy of VMI and TPL can be maximized when the supplier, manufacturer and distributors exchange information with accuracy on right time. Although EDI has been used to facilitate information exchange between them, its expensive VAN rent and implementation cost together with its batch-processing nature has hindered its utilization. However, with the progress of Internet technologies, XML-based open B2B standards have been developed to set a new milieu for VMI and TPL [6]. As the reports regarding the success of RosettaNet e-Logistics applications overseas have been issued, it has attracted interests of Korean enterprises.

To integrate VMI and TPL via RosettaNet, the efficient execution and management of multiple PIPs (Partner Interface Processes) are most important. To achieve this, this study proposes a method to construct a system which monitors the multi-PIP environments, and also presents a method to connect such system with the BPM system through the content based document routing function.

Chapter 2 of this study briefly introduces the RosettaNet e-Logistics program and the BPM systems. VMI processes with TPL are reviewed in Chapter 3. Chapter 4 relates each RosettaNet PIPs to real VMI processes, while Chapter 5 proposes a method to manage multiple PIPs for the VMI process. The final chapter, Chapter 6 summarizes this study and presents future research issues.

2 RosettaNet e-Logistics Program and BPM

2.1 RosettaNet e-Logistics Program

RosettaNet is a non-profit consortium that consists of more than 500 leading companies and organizations working to create and implement industry-wide, open e-business process standards. RosettaNet is also the name of the e-business standard defined by that consortium. It was first launched in June 1998 with about 40 IT companies, but now has grown into one of the most leading e-commerce standard bodies and has extended to include various companies of the electronics, semiconductors, communications and logistics as well as high-tech industry [7, 8].

RosettaNet e-Logistics program covers the overall process of enterprise such as collaborative demand forecasting, order management, shipping, delivery, billing and remittance that are collocated before or after the online transaction between two trading partners. The VMI and TPL are the core processes of this program. In order for it being successful, RosettaNet PIP should be able to support all those processes.

2.1.1 VMI

As a part of e-Logistics program, VMI is related with the business process of collaborative demand forecasting and order management. VMI is used for the producer (the supplier or the wholesaler) to place purchase order to the vendor by utilizing the forecast information of the vendor (distributor or manufacturer). Whereas in the past the vendor of the end product decided deadlines and quantities of the subparts, VMI allows the subpart producers to control the inventory. With the shared forecast information and their own production plan and capacity, subpart producers can efficiently maintain their safety stock level, while removing the subpart inventory control costs of the vendor. Now, the vendor can receive a safer and more accurate inventory service from their reliable producers. Reduced safety stock level, accurate deadlines and reduction of stock-outs may be beneficial to both the vendors and producers [5, 9]. Following figure shows the VMI process between the manufacture and distributor in an EDI environment together with the EDI documents involved [6].

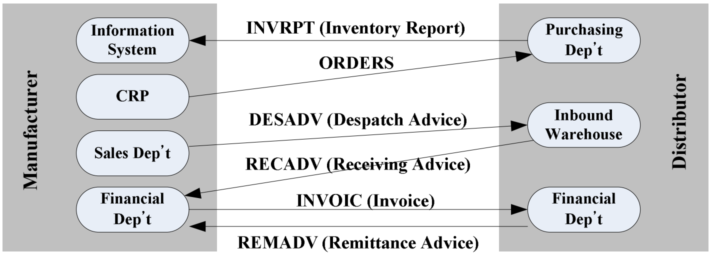


Fig. 1. The Typical VMI Process (*Source: EAN Korea)

2.1.2 TPL

TPL is a company's long-term usage of specialized external logistics service. As the supply chain of the company becomes more complex, there are a number of cases where simple transportation or warehouse management along with customs clearance service, returns of defected goods and packing services are being handled by TPL. In North America or Europe, where the concept of TPL has prevailed, companies apply TPL in various areas with anticipated effects of reduced inventory and transportation cost from specialty, capturing market needs, higher customer services while focusing on their core competences. However, Asia-Pacific enterprises have been passive in applying TPL, lest it should hinder immediate customer response or cause to lose the control over logistics [10].

However, the control over logistic activities can be reinforced by using IT and e-commerce standards. If there is an adequate logistics information system, such fear on TPL can easily be dispelled.

2.2 BPM System and Multi-PIP Support

Today, as the business surroundings such as governmental regulations, policies, and technologies are undergoing rapid changes while customer needs and product requirements become diversified, enterprises make every endeavor to adapt themselves to those changing environment. To increase their competitive power, enterprises are undergoing an endless cycle of process innovation, and try to set up the BPM system in various fields as a supporting tool.

BPM system is an enterprise solution for planning, managing and improving business processes, and is mostly originating from WfMS or EAI [11]. While the BPM system focuses on the enterprise's internal processes, B2B or e-commerce systems focus mainly on the inter-company processes. RosettaNet is an e-commerce standard that can be implemented on the B2B system, and provides PIPs that correspond to inter-company processes.

As the business processes between the trading partners become tightly related, B2B processes that were once separately executed as individual PIPs, now needs to be integrated and totally managed to execute a higher level of process as is done in the e-Logistics program. Although RosettaNet makes mention of PIPs that are relatable, it lacks concrete details on how to realize such connections.

This study implements a content based document routing function that organically connects individual PIPs to build multi-PIP environment like the e-Logistics program. Furthermore, a system that monitors the overall relationship of those PIPs is proposed in this study, together with a method to introduce BPM system for managing the proposed constructs at a higher level.

3 VMI Process with TPL

VMI process can be divided into three stages: demand forecasting and planning stage, inventory and demand fulfillment stage, and invoicing and remittance stage. While conventional VMI process only involved the interactions between the supplier and the manufacturer, VMI process with TPL includes the manufacturer's warehouse and thus covers 3-way interactions between them. The manufacturer's warehouse could be under consigned management by 3rd party logistics company or could be owned by the logistics service company from the very first.

VMI process with TPL can be classified into two types. The distinction is made by whether the demand forecast and planning is determined by the part supplier or the manufacturer (retailer). The former is called the supplier centric VMI while the latter is referred as the demander centric VMI.

3.1 Supplier Centric VMI

In order for the supplier to come up with an adequate demand forecast and plan, the manufacturer provides necessary sales history and inventory status report. The TPL company also needs to periodically report the inventory status of the warehouse, where the quantities on their ways to the warehouse from the supplier and quantities headed to the manufacturer from the warehouse should be included. With such information, the supplier can generate a demand forecast and production plan. The following figure 2 illustrates the supplier centric VMI process with TPL.

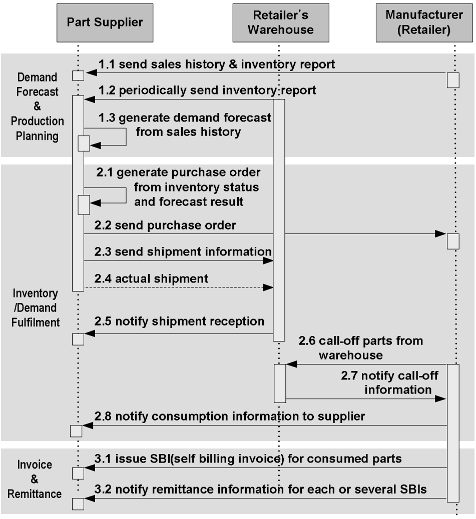

Part Supplier
Retailer's Warehouse
Manufacturer (Retailer)
Demand Forecast & Production Planning
1.1 send sales history & inventory report
1.2 periodically send inventory report
1.3 generate demand forecast from sales history
Inventory /Demand Fulfilment
2.1 generate purchase order from inventory status and forecast result
2.2 send purchase order
2.3 send shipment information
2.4 actual shipment
2.5 notify shipment reception
2.6 call-off parts from warehouse
2.7 notify call-off information
2.8 notify consumption information to supplier
Invoice & Remittance
3.1 issue SBI(self billing invoice) for consumed parts
3.2 notify remittance information for each or several SBIs

Fig. 2. Supplier Centric VMI with TPL

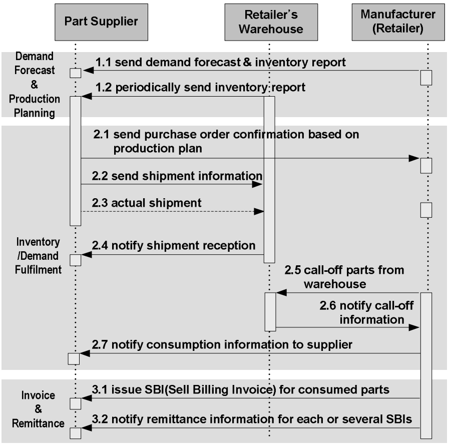

Part Supplier
Retailer's Warehouse
Manufacturer (Retailer)
Demand Forecast & Production Planning
1.1 send demand forecast & inventory report
1.2 periodically send inventory report
Inventory /Demand Fulfilment
2.1 send purchase order confirmation based on production plan
2.2 send shipment information
2.3 actual shipment
2.4 notify shipment reception
2.5 call-off parts from warehouse
2.6 notify call-off information
2.7 notify consumption information to supplier
Invoice & Remittance
3.1 issue SBI(Sell Billing Invoice) for consumed parts
3.2 notify remittance information for each or several SBIs

Fig. 3. Consumer Centric VMI with TPL

In the figure 2, all the processes accompany with the delivery of corresponding PIP documents except the process 2.4 that illustrates actual transportation of products.

3.2 Consumer Centric VMI

In this type, the manufacturer determines the demand forecast and production plan, and then sends the required subpart quantities to the supplier together with its inventory status. The TPL company also report its inventory status to the supplier periodically like did in the supplier centric VMI.

The consumer centric VMI process with TPL is illustrated in the above figure. As in the figure 3, the purchase order between the supplier and the manufacturer is omitted, since the purchase order confirmation sent in process 2.1 replaces the purchase order.

4 Mapping Between VMI Process and PIP

To implement VMI processes with TPL, each detailed processes needs to be mapped to the PIP, and the individual PIPs' data field should be filled with corresponding transaction data. Although the RosettaNet specifies on what data fields are needed within the individual PIP documents, it does not specify on how various PIPs can be integrated together to form a higher level of business processes. This is exactly the reason why enterprises are having difficulty in determining which PIPs to use to implement a real business process. In this study, the following PIPs are selected and then mapped to the individual processes to implement the VMI process with TPL.

The following table contains the detailed list of how PIPs were matched with both the processes of supplier centric VMI and customer centric VMI.

Table 1. Correspondence of VMI process to PIP process

VMI	PIP Process (Supplier Centric VMI)	PIP Process (Customer Centric VMI)
1.1	4E1. Notify of Sales Report 4C1. Distribute Inventory Report	4A3. Notify of Threshold Release Forecast
1.2	4C1. Distribute Inventory Report	4C1. Distribute Inventory Report
2.1	N.A.	4A5. Notify of Forecast Reply
2.2	3A4. Request Purchase Order	3B2. Notify of Advance Shipment
2.3	3B2. Notify of Advance Shipment	N.A.
2.4	N.A.	4B2. Notify of Shipment Receipt
2.5	4B2. Notify of Shipment Receipt	4D1. Notify of Material Release
2.6	4D1. Notify of Material Release	3B2. Notify of Advance Shipment
2.7	3B2. Notify of Advance Shipment	4B2. Notify of Shipment Receipt
2.8	4B2. Notify of Shipment Receipt	N.A.
3.1	3C7. Notify of Self Billing Invoice	3C7. Notify of Self Billing Invoice
3.2	3C6. Notify of Remittance Advice	3C6. Notify of Remittance Advice

As shown in the above table, in the supplier centric VMI, internal processes that are not related to the exchange of business documents (process 1.3 and 2.1) and the process that indicates the physical movement of the product (process 2.4) are excluded in the mapping. It is also important to take notice that two individual PIP processes are used together for process 1.1. The majority of RosettaNet B2B systems regards the basic unit of business process as a PIP and thus assigns a single PIP to manage the process. However, as in the case of process 1.1, when two different PIPs are combined to form a larger business process, there is a certain level of difficulty in managing that particular process. In this study, to overcome such difficulty of managing individual PIPs, RosettaNet B2B system is integrated with the BPM system and thus the overall VMI process could be managed with consistency.

In the customer centric VMI, process 2.3 which represents the physical movement of the product is excluded from the mapping. As for process 1.1, both PIP4A2 (Notify of Embedded Release Forecast) and PIP4A3 (Notify of Threshold Release Forecast) can be used to exchange the demand forecast information. However, in this study, PIP4A3, which indicates that the responsibility of inventory control has been handed over from the manufacturer to the supplier, is used.

5 Multi-PIP Management for VMI

As previously mentioned, in the VMI process with TPL, many PIPs are associated together to form the entire process. Although each PIP is individually executed as a lower level process by the RosettaNet B2B system, each PIP should be organized to form a higher level VMI process. RosettaNet standards, however, does not provide guidelines on how to design the relationship between PIPs and how such PIPs which are under execution can be linked and monitored within the entire process. To model the relationship between such individual PIPs and to manage the entire VMI process with consistency, this study has developed a content based document routing function and a monitoring system. By integrating these with a commercial BPM system, we could control the whole VMI process with TPL easily.

5.1 Mapping Between Process Instances

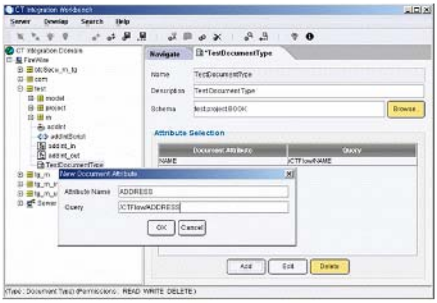
In order for the BPM system to model the entire VMI process, and to execute each PIP transactions through the external RosettaNet B2B system, it is crucial to maintain the comprehensive relationship between the BPM process instance and the individual PIP instances. This is because there is no apparent relationship between the process instances that are initiated independently by the BPM system and the RosettaNet B2B system with the individually assigned process identifiers.

RosettaNet uses 'PIP Instance ID' as an identifier between PIP instances. It is set at the XML element of 'ServiceHeader/ProcessControl/pipInstanceId/InstanceIdentifier' in the PIP document, and its value is to be internally assigned by the party that initiates the PIP transaction [12]. When a PIP document is generated by the internal RosettaNet B2B system and transmitted to the trading partner, the relationship can be managed without difficulty by keeping the generated 'PIP Instance ID' as a 'Process Relevant Data' of the BPM instance that is defined at the WfMC (Workflow Management Coalition) [13]. However, it is quite difficult for the trading partner to bind its running BPM instance to the newly created PIP instance that possesses externally generated 'PIP Instance ID'. Asking for the partner to send one's BPM instance ID in the 'PIP Instance ID' field could be an answer. It, however, can only be achieved if there is a way to deliver one's BPM instance ID to the counterpart. Furthermore, in cases where the internal BPM process instance can only start upon receiving the external PIP document, there would be no BPM process instance ID to deliver at all. Such process ID delivery is not within the scope of RosettaNet standard either.

Another problem that arises from the instance mapping is that the corresponding relationship is not necessarily 1:1, it may be M:N. This means that a single PIP instance could be related with multiple BPM process instances and vice versa. For example, in the supplier centric VMI, process 2.3 (PIP3B2) which forwards the shipment information, and process 2.5 (PIP4B2) which notifies the reception of shipment, may not necessarily be in a 1:1 relationship. The TPL service provider may reply to a multiple number of PIP3B2 documents with a single PIP4B2 document. The opposite case is also possible. In the invoice and remittance stage, supplier could receive several 3.1 process (PIP3C7) documents while receiving a single 3.2 process (PIP3C6) document. This is possible because some enterprises favor remitting at the end of each quarter. In such cases where the relationships are not 1:1, it's hard to relate PIP instance with the BPM instance, and to determine how many PIP documents should be received to advance BPM instance to the next state.

5.2 Content Based Document Routing

To trace the correspondence between the PIP instances and the higher level BPM process instance, a content based document routing function was devised in this study. It is implemented by providing a configuration capability for a group of XPath queries on the PIP document. By executing the queries against a PIP instance and acquiring the results, we can determine what should be done to this PIP instance and thus can specify the relationship between instances. For example, in case of PIP3B2 and PIP4B2 mentioned above, a relationship can be made by extracting the product lot number and container number from the PIP3B2 document and comparing these with those of PIP4B2 document.



Name TestDocumentType

Description TestDocumentType

Schema test.project.BOOK

Attribute Selection

Document Attribute	Query
NAME	/CTFlow/NAME

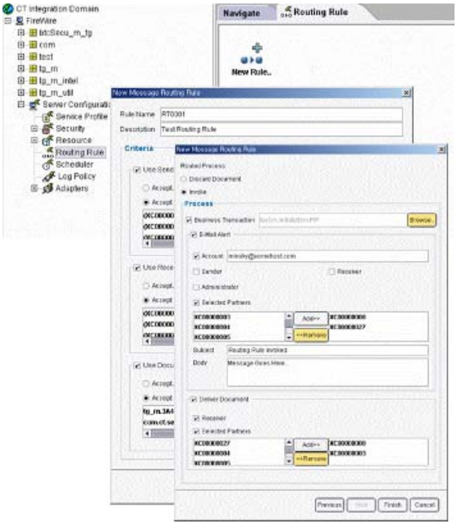
New Document Attribute

Attribute Name ADDRESS

Query /CTFlow/ADDRESS

(Type : Document Type) (Permissions : READ WRITE DELETE)

Fig. 4. Defining the Document Type



New Message Routing Rule

Rule Name RT0301

Description Test Routing Rule

Criteria

Routed Process

○ Discard Document

● Invoke

Process

☑ Business Transaction

☑ E-Mail Alert

☑ Account minsky@somehost.com

☐ Sender ☐ Receiver

☐ Administrator

☑ Selected Partners

Subject Routing Rule invoked

Body Message Goes Here.

☑ Deliver Document

☑ Receiver

☑ Selected Partners

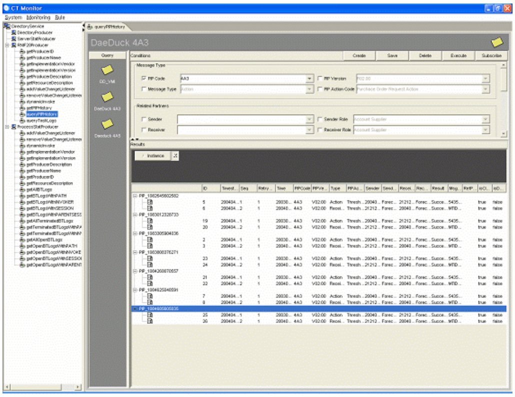
Fig. 5. Defining the Routing Rule

The 'Document Type' and the 'Routing Rule' are defined to realize such content based document routing. A 'Document Type' refers to a group of XPath queries to extract data from the PIP document. In the previous example, to extract the product lot and container number, the following two XPath queries both are declared under the element of 'AdvancedShipmentNotification/Shipment/ShippingContainer' should to be defined as a 'Document Type': '/ShippingContainerItem/traceIdentifier/ProprietaryLotIdentifier' and 'shippingContainerIdentifier/ProprietarySerialIdentifier'. The figure 4 shows the user interface used to define a 'Document Type' in this study.

Once a PIP instance document has been received, XPath queries, defined in the 'Document Type' are executed to extract data from that document. The 'Routing Rule' defines how the retrieved data is evaluated and finally determines how the received document is handled – invoke a corresponding BPM process instance, just send an acknowledgement signal to the sender, etc. The figure 5 shows the user interface to define a 'Routing Rule' in this study.

5.3 Multi-PIP Monitoring

With the content based document routing function, the relationships between the PIPs or between PIP and higher level BPM instance could be maintained. As for the monitoring of the relationship, the runtime status of the BPM process instance can easily be traced through the BPM system, however, the causal relationships between PIPs or between PIP and BPM instance could not be fully monitored by the conventional RosettaNet B2B system or BPM system. Therefore a system that monitors these causal relationships is needed to run the content based document routing function in the production mode. A multi-PIP monitoring system is provided to monitor these causal relationships between PIPs or between PIP and BPM instance in this study. The following picture depicts the multi-PIP monitoring system.



Message Type

☑ PIP Code: 4A3
☐ PIP Version: V02.00
☐ Message Type: Action
☐ PIP Action Code: Purchase Order Request Action

Related Partners

☐ Sender:
☐ Sender Role: Account Supplier
☐ Receiver:
☐ Receiver Role: Account Supplier

	ID	Timest...	Seq	Retry ...	Time	PIPCode	PIPVe...	Type	PIPAc...	Sender	Send...	Recei...	Rec...	Result	Msg...	RefP...	isCl...	isD...
PIP_1082645602582																		
	5	200434...	1	1	20030...	4A3	V02.00	Action	Thresh...	20040...	Forec...	21212...	Forec...	Succe...	5435...		true	false
	6	200434...	2	1	20040...	4A3	V02.00	Recei...	Thresh...	21212...	Forec...	20040...	Forec...	Succe...	MTID...		true	false
PIP_1083012328733																		
	19	200434...	1	1	20030...	4A3	V02.00	Action	Thresh...	20040...	Forec...	21212...	Forec...	Succe...	5435...		true	false
	20	200434...	2	1	20040...	4A3	V02.00	Recei...	Thresh...	21212...	Forec...	20040...	Forec...	Succe...	MTID...		true	false
PIP_1083305904836																		
	2	200434...	1	1	20030...	4A3	V02.00	Action	Thresh...	20040...	Forec...	21212...	Forec...	Succe...	5435...		true	false
	3	200434...	2	1	20040...	4A3	V02.00	Recei...	Thresh...	21212...	Forec...	20040...	Forec...	Succe...	MTID...		true	false
PIP_1083806376271																		
	23	200434...	1	1	20030...	4A3	V02.00	Action	Thresh...	20040...	Forec...	21212...	Forec...	Succe...	5435...		true	false
	24	200434...	2	1	20040...	4A3	V02.00	Recei...	Thresh...	21212...	Forec...	20040...	Forec...	Succe...	MTID...		true	false
PIP_1084260070557																		
	21	200434...	1	1	20030...	4A3	V02.00	Action	Thresh...	20040...	Forec...	21212...	Forec...	Succe...	5435...		true	false
	22	200434...	2	1	20040...	4A3	V02.00	Recei...	Thresh...	21212...	Forec...	20040...	Forec...	Succe...	MTID...		true	false
PIP_1084825840591																		
	7	200434...	1	1	20030...	4A3	V02.00	Action	Thresh...	20040...	Forec...	21212...	Forec...	Succe...	5435...		true	false
	8	200434...	2	1	20040...	4A3	V02.00	Recei...	Thresh...	21212...	Forec...	20040...	Forec...	Succe...	MTID...		true	false
PIP_1084805805835																		
	25	200434...	1	1	20030...	4A3	V02.00	Action	Thresh...	20040...	Forec...	21212...	Forec...	Succe...	5435...		true	false
	26	200434...	2	1	20040...	4A3	V02.00	Recei...	Thresh...	21212...	Forec...	20040...	Forec...	Succe...	MTID...		true	false

Fig. 6. Multi-PIP Monitoring Screen

6 Conclusions and Future Research Issues

This study has analyzed VMI processes with TPL which has been chosen by major global companies as a way of managing their complex supply chains with efficiency, and has also investigated how to implement such processes with the RosettaNet standard. To effectively support VMI processes with TLP, this study has defined and implemented a content based document routing, where the relationships between PIP instances or between PIP and BPM process instance are controlled. Finally, a system that consistently monitors those casual relationships has been constructed too. The system that this study proposes has been successfully implemented by a Korean PCB manufacturing company and runs e-Logistics programs in a production mode with a global mobile phone manufacturing company.

Future research plans to expand the current system include executing complex routing rules with a specialized rule engine to enhance the performance. It is anticipated that additional forms of routing rules can be identified through such future studies.

References

[1] J.C. Tyan, F.K. Wang and T. Du, “Applying collaborative transportation management models in global third-party logistics,” International Journal of Computer Integrated Manufacturing, Vol.12, No.4-5, pp. 283-291, 2003.

[2] K.H. Lai and T.C.E. Cheng, “Supply Chain Performance in Transportation Logistics: An Assessment by Service Providers,” International Journal of Logistics: Research and Applications, Vol.6, No.3, pp. 151-164, 2003.

[3] Jagdev, H. S., and Thoben, K.-D., Anatomy of enterprise collaborations, Production Planning and Control, 12 (5), 437-451, 2001.

[4] B. Rao, Z. Navoth and M. Horwitch, “Building a World-class Logistics, Distribution and Electronic Commerce Infrastructure,” Electronic Markets, Vol.9, No.3, pp. 174-180, 1999.

[5] S.M. Disney, D.R. Towill, “The effect of vendor managed inventory (VMI) dynamics on the Bullwhip Effect in supply chains,” International Journal of Production Economics, Vol.85, pp. 199-215, 2003.

[6] Paraiso, David. Implementing EDI. Macmillan Computer Publisher. 1996.

[7] RosettaNet, RosettaNet Technical Basis, RosettaNet White Paper, <http://www.rosettanet.org/RosettaNet/Doc>, 2004.

[8] RosettaNet, Automating through RosettaNet, Intel Information Technology White Paper, <http://www.rosettanet.org/RosettaNet/Doc>, Jan. 2003.

[9] George Kuk, “Effectiveness of vendor-managed inventory in the electronics industry: determinants and outcomes,” Information & Management, Vol.41, pp. 645-654, 2004.

[10] S.Y. Lee, “A study on the facilitation of Third-Party Logistics in Korea,” Seoul National Univ. Management Graduate School, Dissertation Thesis, 1999.

[11] M. Kim et al. “A Modeling Framework of Business Transactions for Enterprise Integration,” ICCSA 2005, LNCS 3482, pp. 1249-1258, 2005.

[12] RosettaNet, RosettaNet Implementation Framework: Core Specification V02.00.01, <http://www.rosettanet.org>, (6 Mar. 2002).

[13] Workflow Management Coalition, The Workflow Reference Model, TC00-1003, < http://www.wfmc.org/standards/docs/tc003v11.pdf>, (19 Jan. 1995).

Information Security Management System for SMB in Ubiquitous Computing

Hangbae Chang[1], Jungduk Kim[2], and Sungjun Lim[3]

[1] Yonsei University, 134 Sinchon-Dong, Seodaemun-Gu,
Seoul, 120-749, Korea
hbchang@paran.co.kr
[2] Chung-Ang University, 72-1 Nae-ri Daedeok-myeon Anseong-si,
Kyunggi-do, 456-756, Korea
jdkimsac@cau.ac.kr
[3] Yonsei University, 134 Sinchon-Dong,
Seodaemun-Gu, Seoul, 120-749, Korea
sjsjlim@empal.com

Abstract. In this study, an information security management system is developed through theoretical and literary approach aiming at efficient and systematic information security of Korean small and medium size businesses, considering the restrictions of the literature review on the information security management systems and the inherent characteristics of the small and medium size businesses. The management system was divided into the 3 areas of the supporting environment of the information security, establishment of the information security infrastructure, and management of the information security. Through verification by statistical methods(reliability analysis, feasibility study) based on the questionnaire for the specialists, the overall information security management system is structures with the 3 areas, 8 management items, and 18 detailed items of the management system. On the basis of this study, it is expected that small and medium size businesses will be able to establish information security management systems in accordance with the information security policy incorporating the existing informatization strategies and management strategies, information security systems which will enhance existing information management, and concrete plans for follow up management.

1 Introduction

Modern business organizations are investigating a lot of their resources in developing and managing the information systems which link and support resources and operational processes with management strategies and goals in order to improve competitiveness. To this end, while businesses are improving their efficiencies in operation by smooth sharing of information(pro-active features of informatization), on the other hand, unexpected side effects such as information leak in adversity to the expected goals or effects(adverse features). The impact of such adverse functions are becoming a serious problem in that, they not only give partial damage but obstruct and

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 707 – 715, 2006.

deteriorate the development speed of the intellectual information society which has the potential of infinite progress, and weaken national competitiveness in the age of the infinite competition. As a countermeasure, many Koran small and medium enterprises are establishing information security systems at significant cost recognizing the necessity of preventing those adverse side effects of the informatization, however, most of them fail to construct a consistent information security system in compliance with the integral information security management system which should be carried out by a specialized team for information security, but for one-shot introductions.

In order to achieve the goal of the information security investment efficiently, the process of improving by the areas including the awareness of the staffs on the information security, the standard of the established information security system, and the availability of the information security technologies, in accordance with the results of the weakness analyses based on the standard management system of the information security, depart from simple introduction of the information security systems.

In this study, the present situation of the information security systems of the small and medium businesses are surveyed, and the information security system pertinent to the ubiquitous environment which will be implemented in near future is developed on the basis of the survey in order to estimate optimum investment in the information security system of the small and medium business and to provide a control tool suitable for the process.

2 Literature Review

2.1 Characteristics of the Information Security in Small and Medium Business

Due to their lack of resources, small and medium businesses face many difficulties when introducing the policy and system for information security. According to a prior study related to the survey of the present situations of information security in small and medium businesses, is was shown that most of them felt that 'the cost of information security is too much' compared with their business scale and/or fund, which was the biggest problem. next problem was 'insufficiency of accurate standards or practices of information security', and the third was 'difficulty in identifying the objectives of the information security management' which was due to the obscurity in the scope of organization and definitions in the rules.

In order to survey the detailed present situations of the information security of the small and medium businesses, 20 companies had been visited or questioned with letter as king their present progress and problems. The survey results were classified in accordance with their characteristics for identifying the subjects of the information security in small and medium businesses and utilized as the basic data for calculating weighted values

2.2 Studies on Information Security Management System

The Information Security Management System(ISMS) is a series of processes and activities for the implementation of the three objectives of security, described classified hereinabove, by systematically establishing, documenting, controlling and managing the procedures and processes for the improvement of the stability and reliability of the assets of the organization. ISMS is a systematic management system for the safe keeping and controlling of the sensitive information of business, including the information on the human resources, processes and information system, in all.

① BS7799(ISO 17799)
BS7799 was developed under the initiative of the British Board of Trade with the cooperation of major industries including ‘BT’, ‘HSBC’, ‘Marks and Spencer’, ‘Shell International’, and ‘Unilever’, with the title "code of Practice of Information Security Management", for the reference of the managers in charge of the implementation and maintenance of the information security of organization, in the form of a common document for the standardization of the organization. BS7799 was designed to enable mutual confidence between businesses through exchanges by referencing common information security managerial document by providing single point reference enabling the identification of the necessary control measures for the situations being faced by the businesses and by applying it on a wide scope.

② GMITS(ISO 13335)
While the BS7799 had been developed for the security of the overall organization, GMITS(Guidelines for the management of IT Security) is a standard of information security management, with its fundamentals focused on the security of the information systems. The main content of the information security process includes establishing the object, strategy, and policy of information security, performing risk analysis, establishing and implementing IT security management plan, and carrying out through follow-up activities. To describe the information security management, the basic concepts and model were briefly introduced, model was described in the view points of management and planning, and the overall process of the security management of the information system was presented. The document is consisted of technical descriptions, explanations on the control measures and the procedures of selecting the measures in accordance with the security requirements and the specific environments of the organization, and the method of selecting the items of control when linked with external network, e.g., the Internet. The items of control are consisted of 40 items in 7 subtitles of organizational and physical area and 23 items in 5 subtitles of information system area, in 63 items in total. As a standard of information security management in the perspective of the information system security, the GMITS describes security of the information asset, not the overall assets of organization, lacks the definition of the detailed activities and the results of the information security process. Therefore, it is required to develop an information security management system which enables security of the overall assets of organization covering the information security control items which can be quantified in detail.

3 Requirements for the Development of the Information Security Management System for Small and Medium Business

In order to develop information security system for small and medium businesses, the limits in the prior studies were overcome and the characteristics of the information security in small and medium business, and the requirements of the management system development to resolve the problems were clarified.

First, considering that 'since the position of the business is influenced a lot by the economic environment, trade relationship, competitive environments and internal changes, omission and/or simplification of a specific stages should be allowed, continual modification and supplement can be made', it was intended to develop a common information security management system available for any small and medium business. Secondly, considering that 'due to the insufficiency of resources, any small or business which carry out information security may suffer stoppage or delay in sales activities, which will impact severely on the business', control model rather than process model which require more time and cost is suitable, and the standard model should be suggested. Finally, considering that 'due to the insufficient resources to obtain and maintain international standard(BS7799), it should be extendable to international standards in order to enable extension to diverse functionality and to secure external competitiveness', the overall model of level assessment was intended to be designed on the basis of the BS7799.

4 Developing Information Security Management System for Small and Medium Business

4.1 Designing Information Security Management System for Small and Medium Business

Since the level and the strategy of information security in small and medium business should be designed to be suitable for the pertinent level of informatization(Structural Contingency), general structural layers of the business has to be reviewed. The business informatization consisted of the infrastructure of the information technology which is the technological environment and the information supporting environments (IT Support). The infrastructure is consisted of the four basic elements, described bellow, incorporating the physical facilities required to meet the information requirements, management on the computation resources, and services to customers. Technical elements(Component) IT includes computer, printer, database management software, and OS which are from-the-shelf commodities. However, these do not provide meaningful services to customers in the form they were procured. Knowledge and experience elements(Human IT) are the knowledge, experiences, and skills which are produced by combining the technical elements procured in the market in order to provide meaningful services to the information system users. Shared information service elements(Shared IT) includes the service of sharing information partially among the end users of the information system in their job operation. For example, channel management service and information management service are the typical shared information services. Integrated application information service elements(Shared and Standard Applications) are the services provided to and shared by multiple departments such as accounting and personnel information services. Typical example is the ERP((Enterprise Resource Program). Informatization environments are consisted of human resources and system environments(supporting system, training, performance assessment, etc.). Understandings and support of the top management lead the implementation of the strategy of the organization through the allocation of the rare resources and affect the attributes of the organization members. When the infrastructure of the information technology and the informatization environments are established, the Information Audit team will identify the level of rationale of the results from the construction and operation of the information system, proceed with the corrective and preventive activities for the problems identified in the outputs.

In this study, firstly those component elements and the assets for the informatization have been identified, then the scope of information security which will secure those elements are setup to define the information security management system area for small and medium business.

Table 1. Information Security Management System Area Development

Elements of Informatization		System Area
Supporting Environment	Will toward informatization	Information Security Supporting Environment
	Supportive organs	
	Spread of awareness	
	Policy of informatization	
	Investment	
Infrastructure	Technical elements	Establishment of Information Security Infrastructure
	Knowledge & experiential Elements	
	Shared information service elements	
Informatization Audit	Rule observance	Information Security Operation Management
	Response to accidents	

Detailed information security management systems were reviewed in accordance with the information security management system areas defined above and those parts which are not suitable for small and medium business were modified to constitute a detailed information security management system for small and medium business.

4.2 Designing Information Security Management System of Small and Medium Business

In this section, the components of the areas which constitute the information security system of small and medium business, derived by theoretical and literary approaches were defined.

① Supporting Environment for the Information Security
Supporting environments for the information security is the environments which provide tangible and intangible resources required for the effective information security of the organization, which are classified into the establishment of the information security policy(regulations, rules, code), constitution of the information security and defining relevant responsibilities and roles, recognition and will to support the information security of the management(staff), financial investments for the information security, and training of the information security.

② Establishment of the Infrastructure of Information Security

The establishment of the infrastructure of information security means the actual introduction and management of the information security system which is constructed to secure the infrastructure of informatization, consisted of identification and assessment of the information assets(documents, equipments, etc.), risk analysis and assessment(weakness analysis and assessment) of the information assets, response system to the environmental risks(power failure, flood, fire, etc), access control and management of facilities, access control and management of the documents(outputs), human security, access control and management of the network, access control and management of the server, access control and management of the application software, and access control and management of the personal computers.

- Management of the Information Security

Information security management means the management required for the safe and efficient operation of the security measures implemented in the construction stages of the information supporting environment and infrastructure, consisted of the verification of the rationale of the policy and technology, monitoring of the security policy status, maintenance check and change management, response to the security accidents and failure(maintain job continuity).

4.3 Investigation and Analysis of the Information Security Management System of Small and Medium Business

① Sampling and Investigation Method of the Population

The information security management system of small and medium business were constructed with three management system area covering 19 items. Each items, which were finalized through the verification analysis with the specialists questionnaire, have the meaning of the information security management system for small and medium businesses that can identify general tangible/intangible information assets and suggest optimum countermeasures by risk analysis. In order to verify the validity of the management system areas and detail items which constitute the information security management system in small and medium businesses, questionnaire survey was carried out with the specialists who have fundamental understandings on the information security management. Those specialists include the person who are in charge of or have experiences in business information security, those who have experience in training or consultation of information security, those who implement information security management system, and the specialists engaged in the information security business. The questionnaire had been conducted for ten days from Aug. 01. 2005. 58 questionnaires had been collected, of which 52 had been selected and analyzed, excluding 6 incomplete papers. The questionnaire was prepared in 5-point Likert scale to investigate the validity of the management areas and detail items. The importance by business scale had been measured to identify the difference between large and small businesses in the items of the management system.

② Verification of the Study

The priorities of the detail items of the management system of the information security were investigated by the business scale, focusing on the average values with 5 points in full.

In the overall investigation results, the establishment of the information security(regulations, rules, guidelines), information security organization and assignment of duties and roles, and the recognition and initiatives of the management(staff) on the information security have obtained higher marks. This shows that the effectiveness of the supporting environment of the information security is one of the key element of the information security management system and the fundamental of the infrastructure of the information security. On the other hand, the information security training and financial investment in the information security have obtained relatively lower marks, which is understood that it was difficult to grasp the level of constructing actual information security management system.

In accordance with the characteristic difference between the large and small businesses, the importance of the items of the actual management system vary. In small and medium businesses, identification and evaluation of the information assets(documents, equipments), response to the security accident and failure(maintain operational continuity), access control and management of PCs, access control and management of the documents(outputs) showed relatively higher marks to those of the large businesses. This is understood that, since the standards of the system construction of the smaller businesses is lower than those of the larger businesses due to the restricted resources, the priorities are given to the items(identification of the information assets and information security on the personal level) in the basic information security management system. On the other hand, larger businesses construct management systems to secure identified information assets, focusing on the detail items of the management and improvements of the information security system.

Reliability and validity analyses had been carried out to verify the designed information security management system for small and medium business. In this study, prior to the reliability and validity analyses, the management system item of which average mark was less than 3.0, which is the criteria of the desirable validity, was excluded from the analyses. In consequence, the financial investment for the information security drive in the supporting environment management area was excluded.

In accordance with the results of the Factor Analysis, 4 management system items in the information security supporting environment management area were grouped into 2(Information Security Policy, Awareness & Capability of Information Security), 10 management system items in the information security infrastructure construction management area were grouped into 4(Risk Analysis, Physical Security, Managerial Security, Technical Security), and 4 management system items in the information security operational management area were grouped into 2(Observation, Maintenance), defining the information security management system of small and medium business by 3 management system area, 8 management items, and 18 detailed items of the management system.

5 Conclusion

5.1 Results of the Study

In the early stage of information society, emphasis had been focused on the pro-active features of the informatization in order to promote the popularization of information. Recently, the focus has been being shifted to the information security in accordance with the social interest on the problems of the informatization. The necessity of the information security is recognized in most organizations and businesses as well as government agencies and enterprises which have sensitive information. However, for small and medium businesses which lack human and fund resources even for the positive investment in informatization, comprehensive investment in information security is difficult to be performed. Therefore, the methodology of efficient and effective investment in information security with limited resources has been being emphasized.

In this study, an information security management system is developed through theoretical and literary approach aiming at efficient and systematic information security of Korean small and medium size businesses, considering the restrictions of the prior studies on the information security management systems and the inherent characteristics of the small and medium size businesses. The management system was divided into the 3 areas of the supporting environment of the information security, establishment of the information security infrastructure, and administration of the information security. Through verification by statistical methods(reliability analysis, feasibility study) based on the questionnaire for the specialists, the overall information security management system is structures with the 3 areas, 8 management items, and 18 detailed items of the management system.

On the basis of this study, it is expected that small and medium size businesses will be able to establish concrete plans for the information security management systems and information security systems in accordance with the information security policy inclusive of the informatization strategy and management strategy for the preparation of the ubiquitous environment which will be realized in near future.

5.2 Limits of the Study and the Directions of Future Studies

Small and medium businesses are characterized by the inherent features of the trade for their smaller business scale. Therefore, it is required to develop additional component elements of the information security management system which reflect the characteristics of the business types such as information communication, manufacturing(electric & electronics, fiber, chemicals, mechanical, metal, etc.), logistics, finance, consulting, and service. In addition, the methodologies of detailed driving of the information security management system of the small and medium business should be more clearly visualized. Due to the insufficiency of the human resources and time of the smaller business, checklist type profiles of each detail items of the management system should be developed.

Finally, case studies of the implementation of the information security management system developed by the present study in smaller businesses in order for the actual verification. Studies on investigating the progress of the information security stages in smaller business based on the results of the case studies will also be valuable.

References

[1] BSI(U.K), "BS 7799 part1: Information Security Management - Code of Practice for Information Security Management", 1999

[2] Cohen, Fred, "Managing Network Security: How does a typical IT audit work?", Network Security, 1998

[3] Georgios I. Doukidis, Panagiotis Lybereas and Robert D. Galliers, "Information systems planning in Small business: A stages of Growth Analysis" J. Systmes software, 1996.

[4] Gerald Kovacich, "Establishing an information systems security organization", Computer & Security, Vol. 17, 1998

[5] Gupta, M and G. Cawthorn, "Managerial Implications of Flexible Manufacturing for SMEs", (Elsevier Advanced Technology), 1996

[6] ISACA, “Information Security Governance, Guidance for Boards of Directors and Executive Management”, IT Governance Institute, 2001

[7] ISO/IEC: ISO/IEC TR 13335-4: 2000(E), "Information Technology - Guidelines for the Management of IT Security Part 4", 2000

[8] Jan Eloff, Mariki Eloff, "Information Security Management - A New Paradigm", Proceedings of SAICSIT, 2003

[9] Margi Levy, Philip Powell, "SME Flexibility and the Role of Information Systems", (Small Business Economics), 1998

[10] Weill, P. and M. Vitale MIS Quarterly Executive, "What IT Infrastructure Capabilities are needed to Implement e-Business Models?", 2002

[11] XiSEC/AEXIS Consultants, "BS7799 Information Security SME Guide", XiSEC/AEXIS Consultants, 2002

A Study on the Development of Usability Evaluation Framework

(Focusing on Digital TV)

Hong Joo Lee[1], Choon Seong Leem[2], and Sangkyun Kim[2]

[1] School of Literature of Arts, Dankook University, Korea
blue1024@dankook.ac.kr
[2] Department of Information and Industrial Engineering,
Yonsei University,134, Shinchondong, Seodaemoongu,
Seoul 129-749, Korea
leem@yonsei.ac.kr, saviour@yonsei.ac.kr

Abstract. Recently, with the materialization of ubiquitous computing environment, families have begun using the latest electronic products. However, new technology is that there are increasing numbers of products which, while equipped with these advanced functions, fail to have those functions properly utilized because of the difficulties in operating them. Accordingly, one can suggest that having easy operation functions is an important factor for becoming a best seller.

In this paper, an evaluation was carried out on the usability of Digital TV, which can be said to be a representative electronic product used at homes. To have a rational usability evaluation, an evaluation framework was developed by examining and analyzing existing researches and consumer characteristics. By using the results, a usability evaluation was carried out on 100 consumers concerning Digital TVs being sold in Korea, in order to verify its validity.

1 Introduction

Due to the rising quality of consumer awareness in buying products these days and the prevalence of consumer-oriented designs, the phenomenon of products that are inconvenient to use being weeded out is expanding. Naturally, the industries and research organizations have become very interested in designing products that are easy for consumers to operate. The result of such interest led the industries to become more interested in design development that considers the users, and the academic world to publish papers related to various usability issues. It is also true, however, that the industries and research organizations are applying the results of their research directly to the usability evaluation of products without considering the particularities of the products or the needs of consumers. This paper, therefore, considered these matters, developed a new evaluation framework for the usability of electronic

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 716 – 725, 2006.

products, and applied it to the usability evaluation of digital TVs ‘A’, ‘B’, and ‘C’ in order to verify its validity. Digital TV is a representative consumer-type electronic product that consumers use frequently for a long time.

2 Literature Review

2.1 Usability

Usability is currently being recognized as a new dimension in product design [9]. The word ‘usability’ is sometimes used as ‘ease-of-use’. There may be differences in its interpretation for some people, but generally speaking, the word is used to express efficacy of user interface [13]. Interface is the part of a system that users may physically, perceptually or conceptually come into contact with[8]. The human-product interface becomes a medium that accelerates the flow of necessary information when humans use a product. Therefore, when a human-product interactive system or user interface is inferior, users fail to reach the goals of product usage or user capabilities become limited [11].

A broad interpretation of usability is to recognize usability as high-content quality [3]. Lansdale and Ormerod define usability as a collection of attributes that must be evaluated in the interface or that a good interface must have in order to explain product quality [7]. Therefore, in order for a business to gain product competitiveness in today’s extremely fast-changing environment, it needs to newly reorganize itself based on the current concepts of quality design. Although there are studies on the definition of and the elements that compose usability, their results are insufficient to directly apply to product usability evaluation, and they suggest the need for a more technical development for usability evaluation [5, 12].

In order to develop the technique for usability evaluation, there also need to be measurement units that can quantify usability elements and a measurement technique that can measure those units [4,14]. Many kinds of usability measurements and measurement techniques have been examined by scholars like Treu, Dix and Meister, but they are scattered in various literature and are rarely systematically organized [2,10,15]. Accordingly, evaluation measurement techniques must be arranged and classified; measurement units must then be selected for each usability element, units that can adequately quantify the elements in question, and measurement techniques that can efficiently collect each measurement unit must be selected[6]. Usability evaluation evaluates how well a product or system interface agrees with the recognition needs, or the intellectual, physical needs of humans. Because a product’s usability is an issue that arises during the user-product interface interaction process, its evaluation must start from the basic functions carried out by humans in the interaction of human-product interface, in other words, from the analysis of human information processing.

In this paper, based on the results of previous research, and in order to evaluate users and product usability, classified the interaction of user-product interface into four elements: perception, understanding, intellectual decision, and action.

3 Development of Usability Evaluation System

3.1 Development of Evaluation Model

Evaluation model for user interface is composing a task support, usability and Aesthetics. That is easy finding function for user and easy memorizing. Thus, that is easy icon for user interface.

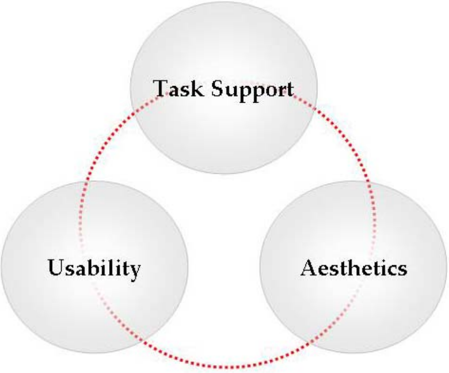


Fig. 1. User Interface Evaluation Model

3.2 Customer Factors for Product-Buying

By using the result, customer important factor for product buying is Table 1.

Table 1. Customer Factors for Product-Buying

Factors	Portion (%)
Marketing Strategy	6
User Convince	17
Design	18
Price	17
After Service	9
Brand	14
Quality of Product	12
Function of Product	4
Etc	3

A user must use a remote controller and buttons on a TV in order to operate it.

Operation of OSD (On Screen Display) is necessary to control TV display and volume, and due to the particularities of the picture medium, connection with an external device is also very important. Consequently, convenience in external connection can be said to be an important element as well.

Accordingly, based on the existing research results of perception / understanding / intellectual decision / action, the four stages of usage process, product usability evaluation elements are proposed as follows.

Table 2. Characteristics of each attribute

Attribute	Characteristics	Importance to Interface
Remote Controller	Wireless device for TV operation	Requires design to enable easy operation for anyone
OSD	TV display operation Software	Requires capability to easily operate device using only icons
External Connection	Convenience in connecting to other devices	Variety of connecting terminals and easy connection
Buttons on Product	Buttons on TV for Operations	Design of placement and size of operation buttons that consider user convenience

This paper examined the numerical index of convenience for each attribute through a survey of 100 users, and produced an evaluation of Digital TV interface classified into four large attributes: remote controller, OSD (On Screen Display), external connection, and operation buttons on the product.

Based on the results of the attributes, an evaluation framework was developed and applied to the usability evaluation of digital TVs currently being sold.
Table 4 is result of important for usability evaluation. The important is major factors using Electronic Products.

Table 3. Importance Level for each attribute

Attribute	Importance Level
Remote Controller Convenience	28.3%
External Device Convenience	24.4%
Buttons on Product	19.8%
OSD Convenience	18.2%
Etc	9.4%

These elements are independent, but they also have mutually complementary characteristics for device operation convenience.

By using Alison's research, this paper is development of evaluation model as follows[1]

Table 4. Detailed evaluation results for each attribute

	Attribute	Attribute Of Digital TV
TASK	Easy finding function.	-Organization Arrangement
	Easy using function	-Connect
	Design help decision making	-Terminals easy to Connect
	Easy Function	Wording Graphics
	Wording Graphics is icon for operation functi on	Wording Graphics
	Design help decision making	Wording Graphics
USABILITY	Easy User Interface	Design
	Easy User Interface for customer memorizing	Organization Arrangement
	Easy User Interface for customer	Usability Button on Product
	User Interface for Correct Operation	Wording Graphics
AESTHETICS	Graphics for Customer Operation	-Wording Graphics -System Selection & Connection
	Wording Graphics for Operation list	-Wording Graphics -Product Design
	Customer is influenced by colorful graphics.	-Design

Table 5. Detailed evaluation results for each attribute

	Attribute	Importance Level
Remote Controller	Easy to use	40.1 %
	Easy to understand	31.7 %
	Design	28.2 %

Table 6. Detailed evaluation results for each attribute

	Attribute	Importance Level
Device Operation	Product Design	48.9 %
	Buttons on Product	28.9 %
	Cleaning management	22.2 %

Table 7. Detailed evaluation results for each attribute

	Attribute	Importance Level
OSD	Understanding / Intuitiveness	33.2 %
	Organization / Arrangement	26.2 %
	System Selection	25.0 %
	Design	15.6 %

Table 8. Detailed evaluation results for each attribute

	Attribute	Importance Level
External Connection	Sufficient Number of Terminals	33.3 %
	Easy to Connect	26.5 %
	Connection Terminal Grouping	18.9 %
	Labels	21.3 %

Based on this, a detailed framework is proposed as follows.

Table 9. Evaluation Framework for Usability

1st attribute	2nd attribute	Items in Detail
Remote Controller	Design	Remote controller size/weight/form, button color/ size/grouping/description, high-class/polish/ easy-to-use appearance
	Usability	Using hands, reaction speed/degree/touch, ease of changing batteries, using primary buttons, using special shortcut buttons
OSD	Design	Menu font/size/background color, high class/polish
	Organization Arrangement	Item classification, sufficient number of items, understanding the procedures, coherence of the procedures
	Wording / Graphics	Wording / understanding the terms, understanding the graphics / pictures
	System / Selection	Operation / selection, sign placement, sign size, sign time, movement speed
External Connection	Connection Terminals	How many external connection terminals?
	Easy to Connect	Labeling / grouping, easy-to-connect Appearance
Device Operation	Buttons on Product	Product button position/size/shape/arrangement/ touch/ distinctiveness
	Product Design	Product casing color/form, speaker color/form
	Cleaning Management	Cleaning, setting up, moving convenience

4 RESULT

Using the evaluation framework, a usability evaluation was carried out for each product, and the result is as follows.

Table 10. Result of Usability Evaluation [Unit : %]

Attribute Product	Remote cont roller	OSD	External Connection	Device Operatio n
A	59.43	78.80	69.94	68.90
B	66.06	76.26	73.64	72.16
C	68.88	58.35	62.78	67.30

A more detailed evaluation of each product's problems is as follows.

Table 11. Satisfaction for Remote Controller [Unit : %]

Classification	Product A	Product B	Product C
Easy-to-use Appearance	62.50	72.90	75.90
Reaction Speed	60.00	69.10	67.70
Understanding Labels	65.90	68.90	67.10
Button Size	54.50	68.80	73.90
Button Grouping	55.00	68.60	65.20
Battery Cover	67.10	68.60	72.30
Remote Controller Weight	52.50	68.20	65.50
Button Shape	51.60	67.30	66.30
Changing Batteries	64.50	66.30	63.80
Using Hand for Operation	52.90	65.90	67.00
Distinguishing Buttons	50.40	65.40	68.40
Button Touch	54.80	64.30	71.60
Button Colors	60.40	63.90	73.20
Reaction Speed	60.50	63.00	69.80
Overall Size	54.50	62.00	63.80
Overall Shape	60.50	62.00	68.00
Polish	70.20	60.40	67.50
High-class Look	70.50	57.00	72.90
Overall Satisfaction	59.40	66.06	68.88

Table 12. Satisfaction for OSD [Unit : %]

Classification	Product A	Product B	Product C
Font Size/Type	86.80	64.10	53.90
Background Color	78.00	70.90	53.30
High-class Look	80.00	73.80	46.60
Polish	80.70	70.70	49.70
Label Position	75.70	72.50	62.60
Label Size	79.80	66.60	66.80
Classification	82.50	77.30	64.60
Inclusion of Necessary Items	75.40	74.80	59.80
Order of Arrangement	80.00	69.80	63.60
Procedure Method	83.00	68.90	55.00
Understand Labels	74.30	70.90	64.80
Graphics	78.00	64.30	58.30
Operation Selection	79.50	70.20	58.30
Sign Time	69.60	65.90	60.80
Movement Speed	70.90	69.90	60.00
Overall Satisfaction	78.80	70.26	58.35

Table 13. Satisfaction for External Connection [Unit : %]

All	Product A	Product B	Product C
Easy-to-connect Appearance	77.9	67.0	64.3
Labels	70.2	68.0	63.0
Terminal Grouping	71.4	69.6	62.5
Sufficient Number of Terminals	60.9	84.3	59.6
Overall Satisfaction	69.94	73.64	62.78

Table 14. Satisfaction for Device Operation [Unit : %]

All	Product A	Product B	Product C
Button Position	70.20	78.90	56.30
Button Shape	67.90	74.80	72.00
Button Touch	67.10	74.50	72.70
Button Distinction	71.30	77.30	56.40
Product Cleaning	75.00	63.00	70.50
Moving the Product	64.30	69.60	65.00
Overall Satisfaction	68.90	72.16	67.30

The usability for each attribute gained through weighting its importance level from [Table 3] is as follows.

4.1 User Convenience = ∑(User Satisfaction * User Importance)

Table 15. Consumer convenience for each product, taking weight into Consideration [Unit: %]

Attribute / Product	Remote co ntroller	OSD	External Connectio n	Device Op eration	Total
A	16.81	19.22	13.84	12.53	20.81
B	18.69	17.14	14.58	13.13	21.18
C	19.49	14.23	12.43	12.24	19.46

Consumers' product preferences were surveyed before the main evaluation in order to verify the validity of the evaluation results, and the results are as follow.

Table 16. Consumer Preference for Digital TV [Unit : %]

	Product A	Product B	Product C
Preferences	32%	39%	29%

Accordingly, it can be said that the validity of the evaluation framework has been verified through [Table 15] and [Table 16] in this paper

5 Issues for Future Research

The Usability evaluation framework proposed in this paper was applied to products currently being sold. Because the proposed evaluation framework was developed with current digital TVs as its subject, in order for it to become an evaluation framework for general electronic products, which are changing all the time, continual research and development is needed.

References

1. Alison, J. Head.: Design Wise: A guide for evaluating the interface design of information resources. cyberage books (1999)
2. Dix, A., Finlay, J., Abowd, G, and Bleale, R.: Human Computer Interaction. Prentice-Hall N.Y (1993)
3. ISO 9241 Ergonomic Requirements for Office Work with Visual Display Terminals Part II.: Guidance on Specifying and Measuring Usability. Committee Draft (1993)
4. Kim, J.W.: Introduction to Human Computer Interaction. An Graphics (2005)
5. Kwahk, J., Han, S.H., Yun, M.H., and Hong, S.W.:Selection and Classification of the Usability attributes for evaluation consumer electronic products. Proceedings of the Human factors and ergonomics society 4th annual meeting (1997)

6. K.Y Park et al.: Usability Evaluation Techniques for the Human Interface of Consumer Electronic Product. Summer Conference on Ergonomics Society of Korea (1997)
7. Lansadle,W.M & Ormerod, C.T.: Understanding Interface. Academic Press Limited (1994)
8. Lea, M.: Evaluating User Interface Design. In User Interface Design for Computer Systems. Ellis Horwood Ltd. 134-167 (1998)
9. March,A.: Usability: The New Dimension of Product Design. Harvard Business Review. 27-32 (1994)
10. Meister,D.: Behavioral Analysis and Measurement methods. John Wiley & Sons (1985)
11. Shackel, B.&Richardson, S.: Human Factors for Informatics Usability. Cambridge University Press (1991)
12. S.M Han et al.:Classification of Usability Elements for the Evaluation of the User Interface of Consumer Electronics Products. Summer Conference on Ergonomics Society of Korea (1997)
13. Stanney, K., Mollaghasemi, M.: A Composite of Usability for Human-Computer Interface Design, In Symbiosis of Human and Artifact Proceedings of the Sixth International Conference on Human-Computer Interaction. Vol.2 387-392
14. S.W Hong et al.: Development of a Usability Evaluation Method. Summer Conference on Ergonomics Society of Korea (1997)
15. Treu,S.: User Interface Evaluation: A Structured approach. Plenum Press (1994)

Designing Aspectual Architecture Views in Aspect-Oriented Software Development*

Rogelio Limón Cordero[1], Isidro Ramos Salavert[1], and José Torres-Jiménez[2]

[1] Department of Information Systems and Computation,
Polytechnic University of Valencia, Camino de Vera s/n E-46071 Valencia - Spain
{rlimon, iramos}@dsic.upv.es
[2] Computer Science Department, CENIDET,
Int. Internado Palmira s/n, col. Palmira. C.P. 62490, Cuernavaca, Morelos, México
jose.torres.jimenez@acm.org

Abstract. Aspect-Oriented Software Development (AOSD) is an area that is becoming important in software engineering. Currently it is focused on how to deal with aspects from the early phases of the software development process, in order to reduce the complexity produced by these aspects in these first phases. Software Architecture (SA) is one of the first steps in the software development process; the SA design requires a support framework to represent, identify, and manage aspects. In this paper, a method to represent and design SA is presented. The proposed method allows: (a) the detection and separation of the architectural aspects and concerns, and (b) the building of the architectural design of the modular and component-connector-aspect views.

Keywords: Sotware Architectures, Aspect Oriented Software Development.

1 Introduction

Aspect-Oriented Software Development (AOSD) is a novel paradigm that aims to improve the separation of concerns [1]. The concerns that (at the design or implementation phases) are spread over several modular units (crossing the limits of these units) are named *crosscutting-concerns*. This crosscutting complicates the development activities, the evolution, and the maintenance of the software. For this reason, a special modular unit that contains the *crosscutting-concerns* (known as *aspects*) was proposed in [1].

With regard to Software Architecture (SA), the aspects involved are called *early aspects* [2]. The identification of *early aspects* may ensure that appropriate decisions are made in the first stages of the software development. However, the techniques to identify and separate aspects are just beginning to be used in an extensive way (not always in an effective way) in the phases of requirement determination and design elaboration [2]. The handling of *aspects* at an architectural level still is in an immature phase.

* This research was supported in part by the **CICYT-PRISMA** project.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 726–735, 2006.

In this paper, a method to generate the aspectual architecture by means of a mechanism of separation of concerns and *early aspects* in a descending and iterative way is proposed. The method is supported by two proposed architectural meta-structures. These allow the representation of the architecture by means of the modular and component-connector-aspect views. With this method, it is possible to design the software architecture and at the same time, identify and specify the *early aspects* of the software development. The method uses the requirements and scenarios given by the users as inputs. The method starts with a tentative high-level architecture and performs a set of activities by breaking the architecture into its elements (functions and components) and establishing the elements' relationships. The aim of the method is to shape the aspectual software architecture by means of the identification and decomposition of the concerns and *aspects*. Furthermore, architectural patterns and a single tool are used, which makes the method more agile. The method is presented using a case study.

The rest of this paper is organized as follows. Section 2 introduces the basic architectural elements in the AOSD context and some related works. Section 3 presents the architectural meta-structures. Section 4 gives an overview of our method. Section 5 develops the method in an actual case study. Finally, section 6 presents some conclusions.

2 Software Architecture and Aspect Design

Software architecture is considered to be a set of system structures [3]. These structures are called views, which include software elements and their external visible properties and the interrelations among them through connectors. The architecture can either be represented in a schematic way or by means of some architectural description language (ADL) for analysis and evaluation proposals.

The architecture design method depends on the approach of the software development process that is followed. For example in the Rational Unified Process (RUP) an object-oriented approach is followed, and the architecture is designed and refined at every stage of the process; it is represented by means of an adaptation of the "4+1" view model.

Another approach proposed in [3] is based on the quality attributes that the software must fulfill; this method is called "Attribute Driven Design" (ADD). The ADD approach can be used with different software development methodologies (for example, the object-oriented and the component-oriented methodologies) where architectural aspects are still barely being considered.

In AOSD, the SA provides a twofold contribution: (a) to deal with early aspect by binding the ones that appear in the requirements with those that are manifested in the design, i.e. aspects of a phase that can reverberate in another phase; and (b) to focus the aspects of the SA taking in account the possible effects on other software development phases.

Thus, to be able to deal with the *aspects* at the architectural level, it is necessary to build a structure that allows for the representation and management of the *aspects*, starting with the separation of the concerns.

Proposals that deal with early aspects appear in [4], [5] and [6]. A method that deals with the software architecture to identify aspects using architectural tactics is proposed in [4]; nevertheless, this method does not deal with the component-connector view. A strategy for aspect decomposition in a top down fashion is proposed in [5]; however it does not deal with software architecture. A method called theme for aspect separation is proposed in [6] (the proposal uses UML notation); however the method does not discuss how to deal with software architecture.

The method proposed in this paper for designing the SA is intended to be part of the AOSD process. The method uses a top-down decomposition strategy and simultaneously makes the aspectual architectural views (in an iterative way), using the nota-tion of UML 2.0.

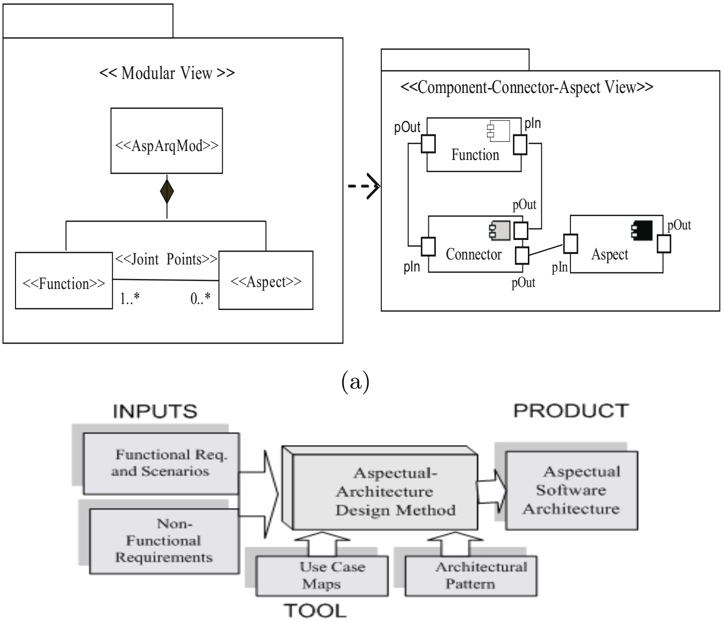
3 Aspectual Software Architectural Views Specification

We propose two meta-structures for dealing with aspectual architectural views, which are based on the architectural description model analyzed in [7]. The intention is to deal with the architectural aspects by means of the two views: the modular and the component-connector-aspect views, which are adaptations of the views analyzed in [2]. Figure 1(a) shows the meta-structures of these views. The first view is used to indicate the system static structure, and the second view expresses the system behavior, which is a realization (represented by the symbol $-- >$) of the modular view.

The modular view contains the *AspArqMod*, *function*, and *aspect* stereotyped classes. The first stereotype is composed of the other two stereotyped classes, i.e. an aspectual architecture is “shaped” by the *function* and *aspect* classes, and their relationship is built by means of an association relation labeled as *Joint Points*.

Since the elements in the modular view are classes (modules) that describe the functional requirements, this kind of module can have its own relationships inside the view such as: decomposition (is a sub-module of), uses, layers, and generalizations [4]. The *aspects* are bound with the functional modules through a prototype relation called *joint-points*, whose cardinality indicates that there are functional modules that do not have a relationship with any *aspect*; but each *aspect* must have at least one relationship with one functional module.

The component-connector-aspect view (C-C-A) expresses dynamism at the component level and has three kinds of components: *functional*, *aspect*, and *connector*. The first two are precisely a realization of the classes with the same name as the modular view. The *connectors* are used to make interactions among the components, and interactions between components and their *aspects*. One of the strategies proposed in [8] is used to specify these components, by means of UML 2.0 notation. This strategy uses a *connector* component type to bind the components through their input (pIn) and output (pOut) ports. An extension of this strategy was required to incorporate *aspects*, which are represented by means of the *aspect* component class; these are bound to functions through *connectors*.



(a)

(b)

Fig. 1. (a) Meta-structures for aspectual architecture, the two proposed views. (b) The elements involved in the architectural design.

The *connector* is a realization of the association relationship of the modular view called *joint-points*. A *connector* can have as many ports as necessary (in accordance with its cardinality) and depending on how many components and aspects are attached by it.

4 An Overview of the Aspectual-Architecture Design Method

The method for aspectual-architecture design is based on three important features: (a) the use of an iterative top-down decomposition strategy, supported by the meta-structures of the aspectual architectural views described previously; (b) the use of architectural patterns; and (c) the use of some behavior analysis tools. The last two are adaptations of the proposals made in [3], [4] and [9].

Figure 1(b) shows the main elements that participate in the method. The inputs come from users and designers; the user contributes the functional requirements, and the designer contributes the non-functional requirements. In this paper, the requirements are expressed by means of use cases, and the scenarios are given in textual form.

A CASE tool based on maps is useful for finding *aspects*, specifying the module behavior, and understanding complex systems [9]. The use of architectural

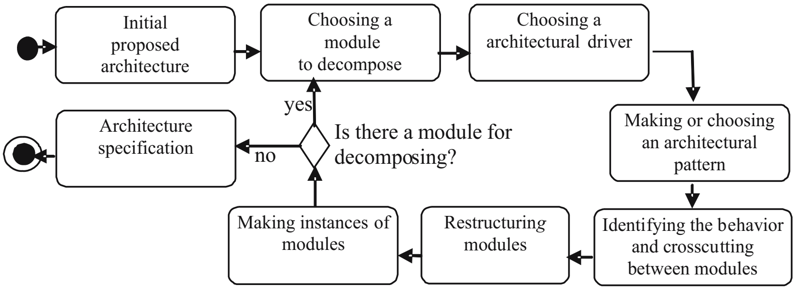

Initial proposed architecture
Choosing a module to decompose
Choosing a architectural driver
Making or choosing an architectural pattern
Identifying the behavior and crosscutting between modules
Restructuring modules
Making instances of modules
Is there a module for decomposing?
yes
no
Architecture specification

Fig. 2. The architecture design method activities

patterns makes the reuse of the architectures possible [3], which simplifies and speeds up the design process. There are some generic patterns, called architectural styles, such as pipe-line and filters, data flows, client-servers, and blackboards, which are the base of many systems.

The separation of concerns proposed in this paper is based on an iterative top-down decomposition process, since this kind of decomposition has proven to be effective in attacking the complexity at the problem-space level. This process allows the functional division of the system and the identification of aspects. The process enables the determination of hierarchical, association, and composition relationships among the functional modules. The joints between the modules and the aspects are also established. The final product of this process is the aspectual software architecture of the system, expressed through its modular and component-connector-aspect views, in UML 2.0 notation. Figure 2 shows the sequence of activities in the method; a description of each activity with an example will be presented in the next section.

5 The Aspectual-Architecture Design Method

5.1 Case Study

We intend to design a system of software for reservation and purchase of tickets in a bus terminal, which has several buses lines; each one of these has a route that passes through different cities.

A typical passenger can perform several functions: make his/her itinerary; choose the city of origin and destination; select a bus line; specify the date and time to travel; and select one or more seats, depending on the availability. All of these activities can be performed through the internet using a credit card or directly at a ticket window at the bus station. For simplicity's sake only one non-functional requirement will be considered; system security, which involves access to users. The user can be a passenger or an employee of the bus company. Security also implies the access to bank accounts to made queries and to apply charges on a credit card (when a passenger purchases a ticket).

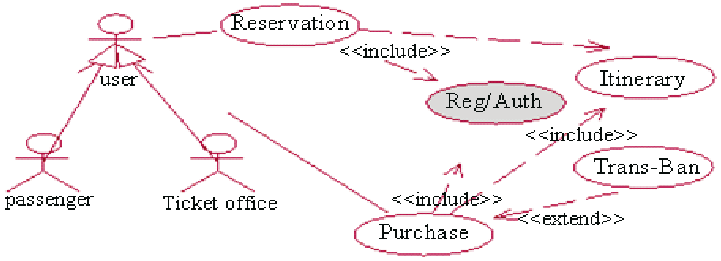


Fig. 3. Use cases for bus ticket reservation and purchase

Table 1. Scenarios Considered

Name	Scenario Description
Purchasing	make itinerary/register and authenticate user / make purchase / validate purchase/ reserve a bus seat/ print tickets
Tickets Reservation	make itinerary/register and authenticate user / make reservation

5.2 Method Inputs

Method inputs are constituted by the requirements and the scenarios; the inputs are given through use cases, as shown in Figure 3, which includes four functional requirements and one non-functional requirement. The figure 3 shows also two types of users, the passenger and the employee of the bus company. The scenarios are described in text form, (see Table 1). For reasons of simplicity only two scenarios are presented.

5.3 Method Activities

Initial proposed architecture. As a starting point, a possible architecture sketch is made, taking into account the specifications given in the functional requirements as well as the non-functional requirements. These requirements are taken at the highest level possible. The initial concerns for the architecture design are derived from these requirements. Figure 4(a) shows the concerns detected in the bus case study. These are only modules, which match a function-stereotype package level in UML.

Choosing a module to decompose. The first time that a module is chosen, any of the modules of the initial proposed architecture can be selected, since they belong to the packet level. In the subsequent times the selection must take into account the unsatisfied requirements with higher priority. In our example, any package can be chosen, since it is the first time.

Choosing an architectural driver. An architectural driver is considered in [2] as “a combination of requirements that shape the architecture or the particular

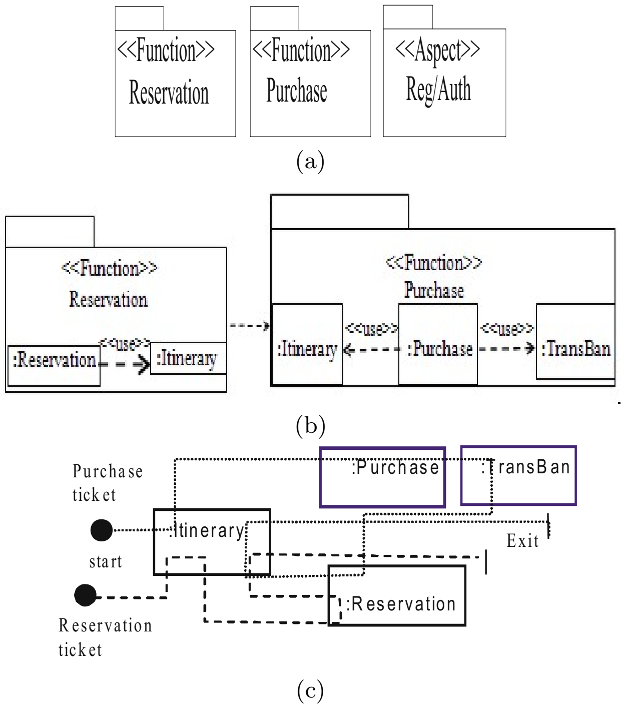


Fig. 4. (a) The initial proposed architecture. (b) Decomposition at the class level, and the application of an architectural pattern. (c) Applying use case maps to the ticket purchase and reserve scenarios.

module under consideration". In our example, we can consider the requirements associated to the purchase and reservation of a ticket, whose scenarios are shown in Table 1. Both of these requirements are used as architectural drivers, and have the same priority. Making or choosing an architectural pattern. An architectural pattern that corresponds or fits the expectation of the architectural driver must be found. This will be used as a base to refine the modules. For the bus example, the architectural drivers previously selected use a client-server pattern for the modules considered above, because the reservation and purchase modules (clients) use the services of the reg/Auth module (server). Next, the reservation and purchase modules are decomposed into sub-modules (classes). The first module will contain the reservation and itinerary classes, creating the ≪use≫ relationship between them. The second module will contain the itinerary, purchase, and transBan classes. The purchase class will use to the itinerary and trans-Ban classes, (≪use≫ relationship), as shown in Figure 4(b).

Identifying the behavior and crosscutting between modules. This task is performed through the use case maps, which use the scenarios as inputs and deal with the modules previously identified, i.e. at the class level. When using the use case maps, a crosscutting can be detected in the two scenarios considered,

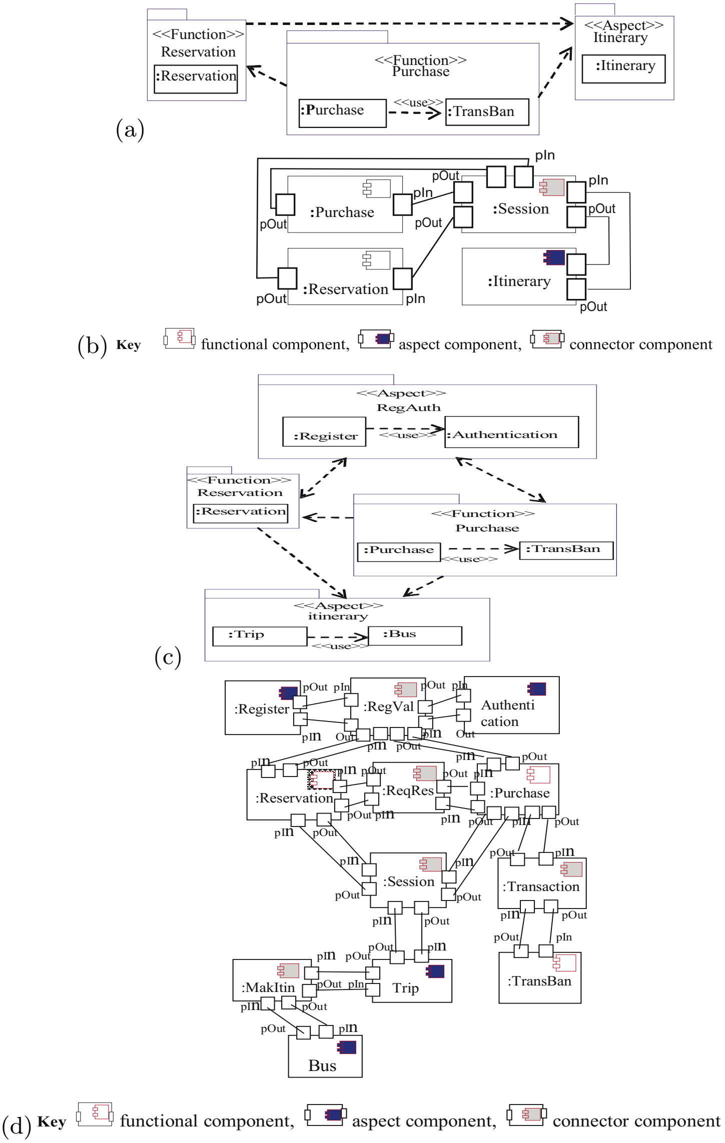


Fig. 5. (a) Modular view of the restructured architecture. (b) Component-connector-aspect view of the restructured architecture. (c) Modular view of the software architecture of the ticket purchase and reservation. (d) Component-connector-aspect view of the software architecture of the ticket purchase and reservation.

since the itinerary class is spread over two classes (*:Purchase* and *:Reservation*). The itinerary class is crossed by the two scenarios, and this class is linked with different classes of distinct modular units (Purchase and Reservation) so the itinerary class becomes a *crosscutting concern*, i.e. an architectural aspect; as shown in Figure 4(c). Now the implicated modules will have to be restructured.

Restructuring modules. When the crosscutting inside a module has been found, it is necessary to change the classes that cause it, moving them to other modules. At the same time the relationship between them is modified. For example, the itinerary class is moved to ≪**aspect**≫ module, as shown in Figure 5(a).

Making instances of modules. In order to make instances for the functional modules, it is necessary to identify their relationships since these relationships must also be instanced. First, the instances of the functional module through components must be made. Then, each relationship must have an instance by means of a special component called a connector. For example, in the bus case, the attachment between the components is made by means of the session connector, as shown in Figure 5(b).

This process continues iterating the activities previously described above (except the first one), until all the modules are defined, as shown in Figure 5(c). Figure 5(c) shows that the modular view contains two functional stereotype packages: the purchase function and the reservation function. The purchase function depends on the reservation function. Both functions contain only atomic modular elements (classes), i.e. modules that can not be decomposed. The functional modules maintain a dependence relationship with the aspectual modules; the regAuth and itinerary *aspects* contain classes used by the functional packages. Figure 5(d) shows the modular view implementation. Each and every class in the modular view corresponds to a component in the Figure5(d). The Figure 5(d) clearly show the relationships between classes contained in the packages through a connector-component type. The information flow in this architectural view is graphically shown by the attachments among the ports (pIn and pOut) of the components. This indicates the dynamic nature of the software architecture.

Architecture specification. At this point, the architectural views have been designed and every module can be specified by means of an ADL. The PRISMA language [10] can be used because it has both a specification language as well as a configuration language; however, this topic is not covered in this paper.

6 Conclusions

This paper proposes a method to deal with *aspects* at the architecture level in the early phases of aspect-oriented development process. The output of the proposed approach is the aspectual-architecture design. The advantages of this approach can be highlighted by the reduction of the complexity inside the problem space, which is this result of the use of a top-down modular iterative decomposition strategy. This strategy separates concerns and *aspects* (*cross-cutting* concerns) in an easy, agile, and effective way, while at the same time composing both the modular view and the component-connector-aspect view, this achieves the aspectual-architecture design. The proposed meta-structures are the framework that allows us to represent, identify, and manage aspects in the software architecture. They also make both a static analysis and a dynamic analysis of the architecture possible. Additionally, the method proposed contributes to: (a) making possible the use of patterns, which allow the architecture to be reused; (b) achieving the traceability between two view types, between the modular view and component-connector-aspect view, and (c) contributing to an improved beginning of the AOSD process.

References

1. R.E. Filman, T. Elrad, M. Aksit M., and S. Clarke. *Aspect-Oriented Software Development.* Addison-Wesley, 2004.
2. A. Rashid, P. Sawyer, A. Moreira, and J. Araújo. Early aspects: Aspect-oriented requirements engineering and architecture design. In *IEEE Joint International Conference on Requirements Engineering*, pages 199–203, 2002.
3. L. Bass, P. Clements, and R. Kazman. *Software Architecture in Practice.* Addison-Wesley, 2003.
4. L. Bass, M. Klein, and L. Northrop. Identifying aspects using architectural reasoning. In *Proceedings Early Aspects: Aspect-Oriented Requirements Engineering and Architecture Design Workshop*, pages 51–57, Lancaster, 2004.
5. C.K. Chang and K. Tae-hyung. Distributed systems design using function-class decomposition with aspects. In *Proceedings of the 10th IEEE International Workshop on Future Trends of Distributed Computing Systems*, pages 148–153, 2004.
6. E. Baniassad and S. Clarke. Theme: An approach for aspect-oriented analysis and design. In *Proceedings of the 26th International Conference on Software Engineering (ICSE04) 0270-5257/04, IEEE*, pages 158–167, 2004.
7. IEEE Architecture Working Group. Ieee recommended practice for architectural de-scription of software-intensive systems. Technical report, IEEE, 2000.
8. J. Ivers and et. al. Documenting component and connector views with uml 2.0. Technical Report CMU/SEI-2004, School of Computer Science, Carnegie Mellon University, 2004.
9. R.J.A. Buhr. Use case maps as architectural entities for complex systems. *IEEE Trans-actions on Software Engineering*, 24(112):1131–1155, 1998.
10. J. Perez, I. Ramos, J. Jaén, and P. Letelier. Prisma: Towards quality, aspect oriented and dynamic soft-ware architectures. In *IEEE Proceedings of the Third International Conference On Quality Software (QSIC'03)*, pages 59–66, 2003.

Process and Techniques to Generate Components in MDA/CB-PIM for Automation*

Hyun Gi Min and Soo Dong Kim

Department of Computer Science, Soongsil University,
511 Sangdo-Dong, Dongjak-Ku, Seoul, Korea 156-743
hgmin@otlab.ssu.ac.kr, sdkim@ssu.ac.kr

Abstract. Component-Based Development (CBD) is an effective approach to develop software effectively and economically through reuse of software components. Model Driven Architecture (MDA) is a new software development paradigm where software is generated by a series of model transformations. By combing essential features of CBD and MDA, both the benefits of software reusability and development automation can be achieved in a single framework. In this paper, we propose a Component-based P Platform Independent Model (CB-PIM) and a UML profile for specifying component-based design in MDA framework. We suggest mapping rules to transform CB-PIM into Platform Specific Models (PSM). Once components are specified with our profile at the level of PIM, they can be automatically transformed into PSM and eventually source code implementation.

1 Motivation

Model Driven Architecture (MDA) is a new software development paradigm where a model plays a key role in automatic software development [1]. It provides a systematic framework to understand, design, operate, and evolve all aspects of an enterprise system, using engineering methods and tools. The framework is based on modeling different aspects and levels of abstraction of such systems, exploiting interrelationships between these models.

A common technique for achieving platform independence is to target a system model for a technology-neutral virtual machine. A model in Platform Independent Model (PIM) is reusable over different platforms. Hence, we regard PIM as neither an executable nor implemented unit. PIM makes traceability among models and improves maintainability through modifying model and regeneration into a Platform Specific Model (PSM).

Component-Based Development (CBD) is another promising approach to develop software system effectively and economically through reuse of software components. Especially, domain-common components provide a common set of features and

* This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD). (KRF-2004-005-D00172).

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 736 – 745, 2006.

functions in a domain, so that application members can utilize the components by customizing the behavior with minimum effect.

We strongly believe that integrating MDA with CBD could yield a unified and effective software development framework, where domain commonality and variability (C&V) are modeled and developed as MDA compatible components, and the software can potentially be generated by transformations. Moreover, both PIM-level components with C&V as well as code-level components can be reused for different platforms.

In this paper, we suggest techniques to combine and maximize the advantages of CBD and MDA. We propose a Component based PIM (CB-PIM) and a UML profile for specifying component design. The profile consists of UML extensions, notations, and related instructions to specify elements of CBD in various MDA models. We also suggest a process to generate diverse components that are implemented by each component model such as Enterprise JavaBeans (EJB), Common Object Request Broker Architecture (CORBA), and Java. Mapping rules to transform CB-PIM into PSM for automation are presented. Components developed with our proposed approach can greatly increase reusability and effectiveness of code generation by transformation.

2 Foundation

2.1 Model Driven Architecture (MDA)

MDA is an approach to using models in software development. The essence of MDA is making a distinction between PIMs and PSMs. To develop an application using MDA, it is necessary to first build a PIM of the application, then transform this using a standardized mapping into a PSM, and finally map the latter into the application code by automation.

The three primary goals of MDA are portability, interoperability and reusability through architectural separation of concerns [1]. Some of the motivations of the MDA approach are to reduce the time of adoption of new platforms and middleware, primacy of conceptual design, and interoperability. The MDA approach makes it possible to save the conceptual design and the MDA helps to avoid duplication of effort and other needless waste [2].

2.2 UML Profile

A UML profile defines standard UML extensions that combine and/or refine existing UML constructs to create a dialect that can be used to describe artifacts in a design or implementation model. The UML profile defines a set of UML extensions that capture the structure and semantics. It defines several standard extension mechanisms, including stereotypes, constraints, tagged values and icons [3]. When one defines a profile, it is common MDA practice to also define mappings that specify how to transform models conforming to the profile into artifacts appropriate to the kinds of systems. If a model is not specified by a particular UML profile, the model can not be transformed automatically by the MDA mechanism.

The Object Management Group (OMG) has adopted a Meta Object Facility (MOF) metamodel of Java and EJB to complement the UML profile for EJB, a UML profile for modeling enterprise application integration and a UML profile for CORBA as well. However, the profiles only support implementation levels and do not represent components of a PIM level

3 Concept of Component-Based PIM

The software development process of MDA is driven by the activity of modeling a software system. The MDA defines the PIM and PSM, creates code, and also defines how these relate to each other. When a designer designs a PSM in previous MDA, the designer reuses a PIM that includes domain information which platform independent information, and writes general component information and specific component platform information for the platform.

If the component platform is changed, the designer rewrites general component and specific component platform information as shown in Fig. 1(a). The specific component information can be reused. However, the previous PSM already includes general component information. The general component information of PSM cannot be reused.

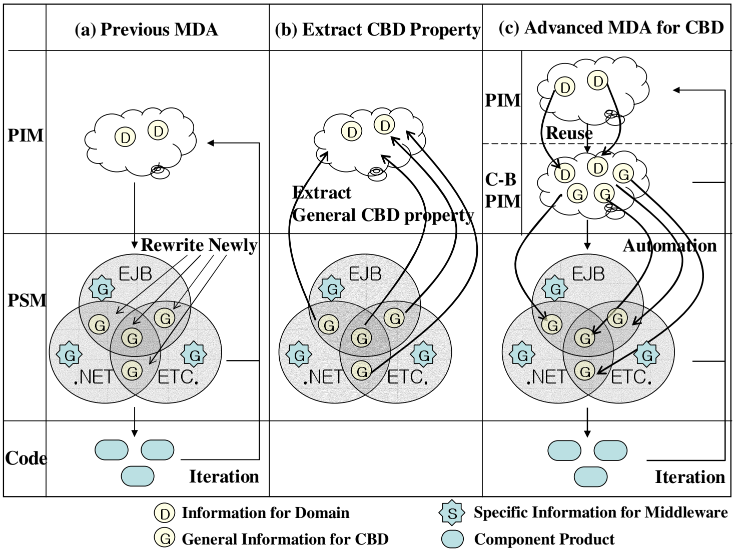


Fig. 1. Advanced CB-PIM using UML Profile for CBD

The PIM is a model with a high level of abstraction that is independent of any implementation technology. However, the PIM that is added to general component information depends on a CBD projects. This PIM for CBD is called a component based PIM (CB-PIM).

The designer rewrites general component information as in Fig. 1(a). It is very redundant work. The work can involve reuse of a previous model. To exclude the redundant work, the component information should be abstracted as in Fig. 1(b). If this information is extracted to a high level, the abstracted design model increases reusability. A designer reuses the abstracted general component information and writes only the new specific component information for a new component platform. If the PIM includes the general component information, the PIM can transfer into PSM automatically as Fig. 1(c).

The UML 2.0 does not cover the entire element in CBD. Therefore, we need a UML profile for specifying components in the CB-PIM level. However, the MDA tool cannot transfer CB-PIM to PSM because the OMG does not have the standard UML Profile for CBD.

4 Methods to Specify Components for CB-PIM

In this section, a UML profile for specifying components is suggested. Our UML profile to represent CB-PIM is based on the UML 2.0. Some elements that are supported by UML 2.0 [6] are used in our profile; the element for CB-PIM that is not supported by UML 2.0 is extended from MOF. The UML profile is compliance with MOF. Therefore, the CB-PIM that is specified by the suggested profile can be presented by common MDA tools.

4.1 UML Profile for Specifying Component Units

In CBD, a component is the fundamental unit of packaging related objects [5], hence we need to specify the related objects in a component in PIM. A port is a connection point between a classifier and its environment. Connections from the outside world

Table 1. The Elements of UML Profile for Component Units Design

Element	Presentation	Applies to	Remarks
Component	«component»	component	Use UML 2.0
System Component	«sysComponent»	component	
Business Component	«bizComponent»	component	
Transient Class	«transient»	class	
Persistence Class	«persistence»	class	Default
Primary Key	«uniqueId»	attribute	
Synchronous Message	«sync»	method	Default
Asynchronous Message	«async»	method	
Message Call	«use», «call», etc.	dependency	Use UML 2.0
Relationships	association, inheritance, composition, aggregation , dependency	relationship	Use UML 2.0
Constraints	{ }, pre:, post:, inv:	class, method, re lationship, etc.	Use OCL
Algorithm	Use Text	method	Use OCL, ASL

are made to ports according to provided and required [6]. Workflow in a component can be designed by sequence and communication diagrams according to UML 2.0. The UML profile for specifying component units is presented as Table 1.

4.2 UML Profile for Specifying Interfaces

A component provides its component-level interface, i.e. the protocol for accessing the service of the component. In CBD, an interface is clearly separated from component implementation to increase the maintainability and replaceability [5]. Hence, we need to specify some interfaces as well as component units in CB-PIM as in Table 2.

Table 2. The Elements of UML Profile for Interface Design

Element	Presentation	Applies to	Remarks
Interface	«interface»	Interface	Use UML 2.0
Provided Interface	«providedInterface»	Interface	Use UML 2.0
Customize Interfac e	«customizeInterface»	Interface	
Required Interface	«requiredInterface»,	Interface	Use UML 2.0
Signature	operationName(param:Type): ReturnType	Operation	Use UML 2.0
Constraints	{ }, pre:, post:, inv:	Class, Method, Relationship	OCL
Algorithm	Use Text	Method	OCL, ASL

In CBD, three types of interface can be modeled; provided, customize and required interfaces. The provided interface specifies the services provided by a component and it is invoked by other components or client programs at runtime. The stereotype «providedInterface» is used to denote this interface. Components often provide mechanisms to tailor the behavior of the components through an interface designed especially for this purpose. A customize interface consists of methods that are used to assign a variant to a variation point [4]. The required interface specifies external services invoked by the current component, i.e. a specification of external services required by the current component. By specifying the required interface for a component, we can precisely define the services invoked by the current component.

4.3 UML Profile for Specifying Variation

The commonality and variability is made explicit through variation points and variants in the components and other reusable component elements [5]. The goal is to create a set of reusable components that expresses commonality and variability appropriate to the family of applications.

The variability can increase the reusability of component. However, the UML does not support notations of variability. Therefore, variability is designed by non-standard stereotypes, tagged values, or note elements [7]. Types of variation are defined as attribute variability, logic, workflow, persistency and interface variability as in [4]. To express variation points of a component in CB-PIM, «vp-Attr», «vp-Logic», «vp-WF», «vp-Persistency» and «vp-Interface» stereotypes are proposed, as in Table 3.

Table 3. UML Profile for Variation Design

Element	Presentation	Applies to	Remarks
Variation Point (VP)	«vp»	Attribute, Method	
Attribute VP	«vp-Attr»	Attribute, Use case	
Logic VP	«vp-Logic»	Method	
Workflow VP	«vp-WF»	Method	
Interface VP	«vp-Interface»	Operation	
Persistency VP	«vp-Persistency»	Operation, Method	
Variant	«variant»	Class, Operation, Method	
Variation Scope	{vScope = value}	Variation Point	Close, Open
VP ID	{vpID = value}	Variation Point, Variant	
Variant ID	{varID = value}	Variant	
Constraints	{ }, pre:, post:, inv:	Class, Method, Relationshi p	OCL
Algorithm	Use Text	Class, Method	OCL, ASL

Fig. 2 shows an example of expressing variability in CB-PIM. The logic of *calculateintereste()*can be changed by each family member. The class 'LoanApplication' has two variation points which are a guarantor and replyCount. Two variants of the attribute guarantor are a type String and a class Guarantor.

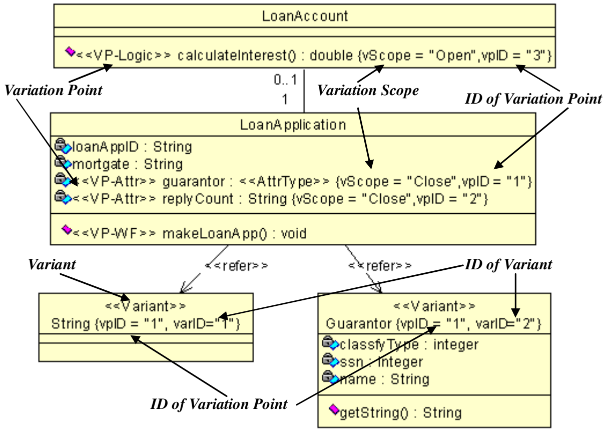

LoanAccount
<<VP-Logic>> calculateInterest() : double {vScope = "Open",vpID = "3"}
Variation Point
0..1
1
Variation Scope
ID of Variation Point
LoanApplication
loanAppID : String
mortgate : String
<<VP-Attr>> guarantor : <<AttrType>> {vScope = "Close",vpID = "1"}
<<VP-Attr>> replyCount : String {vScope = "Close",vpID = "2"}
<<VP-WF>> makeLoanApp() : void
Variant
<<refer>>
<<refer>>
ID of Variant
<<Variant>>
String {vpID = "1", varID="1"}
<<Variant>>
Guarantor {vpID = "1", varID="2"}
classfyType : integer
ssn : Integer
name : String
getString() : String
ID of Variation Point

Fig. 2. Example of Expressing Variation

The attribute guarantor has variation with two variants; String and object Guarantor. If the variant string is set as {varID="1"}, the attribute has string data type to store a guarantor's ID. If the object Guarantor is set, the data type of the attribute becomes the Guarantor. In the implementation process, the variation will be implemented by the value of varID later.

5 Component Development Process Using CB-PIM

In this section, we propose a component development process using CB-PIM and UML profile for specifying components to improve the applicability of PIM of the component level as in Fig. 3. The analysis process extracts functional and non-functional requirements. The analyzed requirements are represented using UML 2.0 by object oriented design process. This process yields PIMs based on objects.

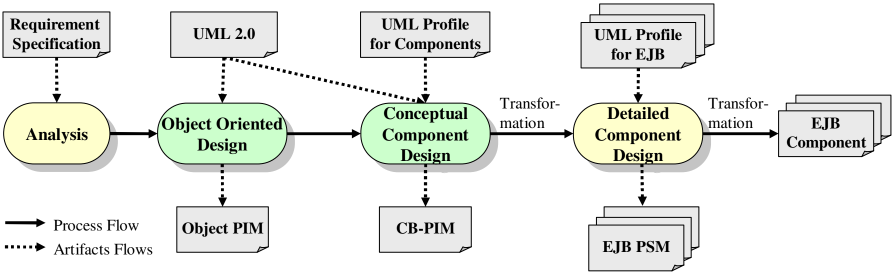


Fig. 3. Component Development Process using CB-PIM

In the conceptual component design, the PIMs of object level transform into component-based PIM (CB-PIM) that presents general component information. The general component information that is units, interfaces, variability, and environments of components does not depend on component platforms such as EJB, CORBA, etc. This process identifies the general component information. None of these can be represented by UML 2.0. Therefore, we need a UML profile for specifying components to present these. The UML profile will be introduced later. Object PIM transforms into CB-PIM that is not dependent on component platform such as EJB and CORBA.

In the detailed component design, the CB-PIM can be automatically transformed into each PSM using the UML profile for component platforms such as UML profile for EJB. Finally, the generated PSMs are transformed into each component source. Therefore, traditional MDA process reuses the object level of PIM. The suggested MDA process reuses the component level of CB-PIM. Once components are specified with our profile at the level of PIM, they can be automatically transformed into PSM and eventually source code implementation.

6 Methods to Generate Components of Multi-platforms

In this section, we suggest a method to generate components that are based on diverse component model such as EJB and CORBA. The method can be supported using tools for automation.

6.1 Method Using CB-PIM and UML Profile for General Components

If components are specified with the suggested UML profile for specifying components at the level of CB-PIM, source codes for each component model can be automatically generated as in Fig. 4. A CB-PIM can be reused and put into diverse platforms.

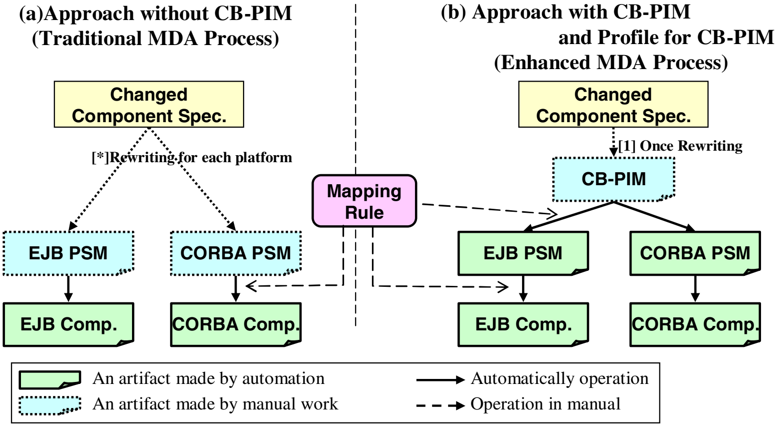


Fig. 4. An advantage of CB-PIM and UML Profile for Specifying Components

In the traditional paradigm without MDA, components for each platform are implemented by coders. Therefore, the paradigm needs many coders for each platform such as EJB or .NET framework. When requirements are modified, designers redesign diagrams for each platform and developers re-implement each component manually.

In the traditional MDA paradigm without CB-PIM as in Fig. 4(a), when requirements are modified, PSMs for each platform are re-designed by designers manually. The PSM can be automatically implemented by MDA.

In the MDA paradigm with CB-PIM as in Fig. 4(b), when requirements are modified, a CB-PIM is changed by a component designer. The CB-PIM can be automatically mapped into PSMs for each platform using mapping rules. The PSM can be automatically implemented by MDA.

6.2 Mapping Rule

A UML profile defines standard UML extensions that combine and/or refine existing UML constructs to create a dialect that can be used to describe artifacts in a design or implementation model. If the model is not specified by a particular UML profile, the model can not be transformed automatically by MDA mechanism.

The CB-PIM should conform to UML profile for CBD (in section 4) because the mapping rule describes relationships between elements of CB-PIM and UML profile. The mapping rule refers to UML profile for CBD. The elements of CB-PIM map into the elements of PSM for EJB and CORBA using UML profile for CBD as Fig. 5. The PSM profiles have been supported by OMG.

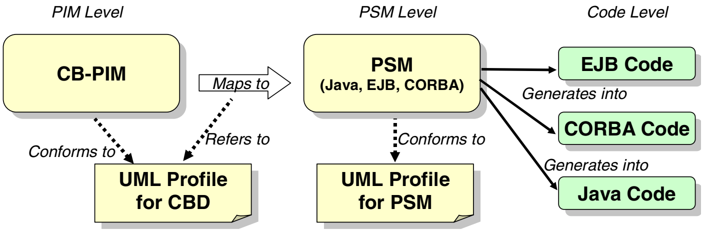


Fig. 5. Mapping CB-PIM into PSM for Specific Components

For example, «persistence» attributes in CB-PIM model map into attributes of an entity bean. «uniqueID» attributes in CB-PIM model map into attributes in key classes for the entity bean. When components are assembled, the conformance between components can be verified by the «requiredInterface» elements in CB-PIM.

6.3 Prototype

If this paradigm for implementing components is supported in a tool, various components can be implemented effectively using seamless method and tools. Eclipse which basically includes a code editor, has plug-in mechanisms, and can plug into modules such as the component designer prototype and code generator as in Fig. 6. Therefore, components are specified with UML profile for CB-PIM, and can be automatically transformed into each PSM and eventually each source code implementation.

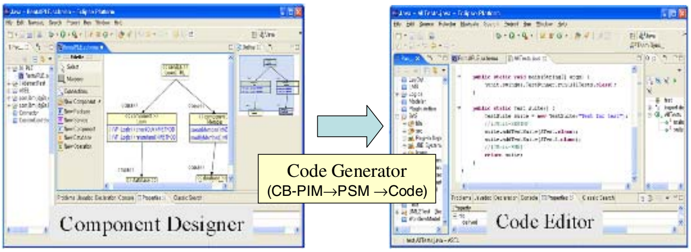


Fig. 6. Tool based on Eclipse to Generate Components for Multi-platform

7 Conclusion Remarks

By reusing software components, software systems can be effectively and economically developed. Especially, domain-common components provide a common set of features and functions in a domain, so that application members can utilize the components by customizing the behavior with minimum effect.

MDA is a new software development paradigm where a model plays a key role in automatic software development. A model in PIM is reusable over different platforms. Hence, we regard PIM as neither an executable nor implemented unit. PIM makes traceability among models and improves maintainability through modifying model and regeneration into a PSM.

In this paper, we suggest techniques to combine and maximize the advantages of CBD and MDA. We propose a CB-PIM and a UML profile for specifying component design. We also suggest a process to generate diverse components that are implemented by each component model such as EJB, CORBA, and Java. Mapping rules to transform CB-PIM into PSM for automation are presented. Both PIM-level components with C&V as well as code-level components can be reused for different platforms. Components developed with our proposed approach can greatly increase reusability and effectiveness of code generation by transformation.

References

[1] OMG, “MDA Guide Version 1.0.1,” *omg/2003-06-01*, June 2003.

[2] Frankel, D. and Parodi, *The MDA Journal, Model Driven Architecture Straight from the Masters*, Meghan-Kiffer Press, 2004.

[3] Frankel, D., *Model Driven Architecture™:Applying MDA™ to Enterprise Computing*, Wiley, 2003.

[4] Kim, S., Her, J., and Chang, S., “A theoretical foundation of variability in component-based development,” *Information and Software Technology, Vol. 47*, *p.663-673*, July, 2005.

[5] Heineman, G. and Councill, W., *Component-Based Software Engineering*, Addison Wesley, 2001.

[6] OMG, *Unified Modeling Language: Superstructure Version 2.0*, ptc/03-08-02, 2003.

[7] Geyer, L. and Becker, M., “On the Influence of Variabilities on the Application-Engineering Process of a Product Family,” *SPLC2 2002, Lecture Notes in Computer Science Vol. 2379*, 2002.

An Ontology Definition Framework for Model Driven Development

Yucong Duan[1], Xiaolan Fu[2], Qingwu Hu[3], and Yuqing Gu[1]

[1] Institute of Software, Chinese Academy of Sciences, Beijing 100080, China
duanyucong@263.net, guyq@sinosoftgroup.com
[2] State Key Laboratory of Brain and Cognitive Science,
Institute of Psychology, Chinese Academy of Sciences, Beijing 100101, China
fuxl@psych.ac.cn
[3] China Development Center for NEC Telecom System,100044, China
huqingwu@bjronghai.com

Abstract. Ontologies are increasingly adopted to facilitate the model driven software development (MDSD). The issue of intertransformations among various ontologies is quickly growing prominent. Some explain that variation of ontology definition should be accepted as a must. We would agree with the point that the definition of ontology for the MDSD should be further standardized with tools. In this paper, the ontology definition for the MDSD is systematically analyzed from the philosophical and human cognitional views. Based on a few metaconcepts, ontology creation and evaluation modes are proposed. By providing general precise and consistent semantics for development elements, this framework will considerably improve the development of models of automation oriented development MDSD processes. Experimental applications on intertransformations and unifications of semantics of existing modeling languages are very encouraging.

1 Introduction

Although definitions of ontologies are still not unified [17], the advantages of ontologies in the model driven software development (MDSD) have been widely acknowledged. It is increasingly adopted to facilitate MDSD. Generally linked ontologies provide applications with shared terminologies and understanding [28]. In the development lifecycle of MDSD, different phases will involve quite different and relatively [24] isolated concept/element sets as is obvious in e.g. common UML practices.

- Statically the cognition of the concepts/element set is in fact one of the original sources of development complexities which involves matters of the concept/element semantics precision, overlap, inconsistency and is in direct ratio with the cardinality of a set.
- Dynamically [30] the combination of the "introduction → disappearance" pattern of various concepts/elements which is determined by their individual lifecycles seems to contribute partially to the "fluctuation" of cognition complexity.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 746 – 755, 2006.

Both problems have been the focuses of many researches and applictions for a long time. Orignizations such as W3C are always underway of efforts on providing the standards or formats in which ontologies are represented such as the Web Ontology Language (OWL), SWRL and SCL. Ontologies have been well acknowledged to be efficient in reducing the static complexity mentioned above. The dynamic problems are usually studied with various model transformation (MT)[2], [4], [16] approaches which start from the semantics mapping and construction of bridging structure of existing concepts/element sets. These ontological concepts/element sets are defined subjectively [1], [16] by human with few commonly accepted uniform creation modes, evaluation criteria. This situation has been most noticeably worsened by mixed Weltanschauungs [19]. The sets can lag far behind the ideal conditions in aspects of concise, precise and efficiency, etc. Most MTs applications will consist of conventional components developed in languages such as Java, and comprehensive methodologies are needed that integrate these components in a real context. Therefore these MTs are quite complex and could be more aesthetical generally. This can be partially attributed to that ontologies have not achieved a major breakthrough yet [28]. Also the issue of intertransformations among various ontologies [21] is quickly growing prominent. We would also agree with the point that the definition of ontology for the MDSD should be further standardized [13], [18].

While a lot of effort has being devoted to defining ontology more standardized, they always focus on providing languages and appropriate tool support directly. Work on development and evaluation methodologies for ontologies is still in its infancy. The breakthrough seems to calls for the emerging a fundamental infrastructure to support the definition and evaluation of these ontologies. It should take full considerations of MDSD relevant issues to better optimize the currently languages, tools and other facilities. Based on in depth analysis of cognition phenomena of ontologies creation with regard to MDSD, a mechanism for ontologies creation and evalutation modes is proposed to fill the absence mentioned above.

In this paper we will explore some issues of the ontology definition for the MDSD from the philosophical and human cognitional views. Based on a few metaconcepts, ontology creation and evaluation modes are proposed. The rest of the paper is structured as follows. MDSD relevant concerns and metaconcepts are introduced in Section 2. Based on analysis on the whole lifecycle of cognition phenomena related to MDSD, Section 3 proposed a philosophical ontologies cognition framework which contains ontologies creation and evaluation [20], [22], [23], [26] modes. Section 4 shows the application of the modes. Section 5 is the related work. The last section ends with the conclusion and future work.

2 Concerns and Metaconcepts Introduction

2.1 Main Philosophy and Concerns

Philosophy. All things in the world of thought with regard to MDSD are connected. This means that things are related or connected or even separated with general meaning relationships in MDSD.

To improve automation of MDSD, all relevant relationships should be found and labeled out. The preposition is that we have clear and correct understanding of the semantics of the models. In fact, this mission is quite complex as it involves all imaginable problems and inconsistent or even conflicting concerns of various stakeholders and development phases along development lifecycles. By investigating on MDSD modeling processes, we discerned the following cognitive concerns: automatic vs. manual, computable vs. uncomputable, deterministic vs. nondeterministic, objective vs. subjective [16], implemental vs. descriptive, etc. At a high abstract level, we can ontologically classify the relationships of these concept couples with "automatic vs. manual" as shown in Fig. 1.

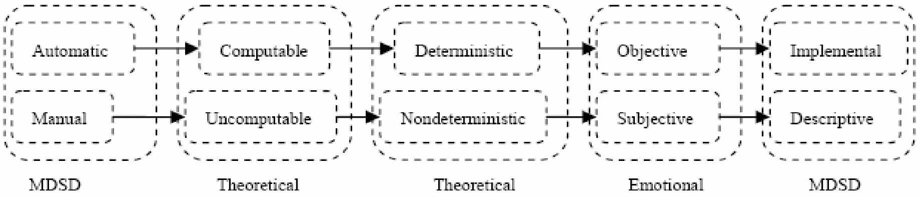


Fig. 1. Main cognition concerns of MDSD modeling

If semantics are given individually for each elements isolated, from different concerns or viewpoints, largely different semantics can be derived. Ross [12] gives that "development process = concepts set" and that processes should be expressed explicitly. So we partially agree with the opinion that currently the main problems of software engineering are rather conceptual than technical [11]. Many model concepts clashes and confusions [8], [27] are found or introduced in the modeling area. One of the consequences of these phenomena is that sometimes the solutions have to be tricked and trapped into the understanding awkwardness created by the awkwardness in the understanding of the problems themselves. This is similar to the language problem which is once pointed out in the area of philosophy by Ludwig Wittgenstein [6].For example, in OOP the attribute is explained as "a property that describes an object or a class" [5]. While in the ER model, it is explained as a function [10]. In Semantic Web languages, such as RDF and OWL, a property is a binary relation [7]. More examples are "attribute vs. property vs. slot", "instance vs. object", "type vs. class", "relationship vs. operation", "aggregation vs. composition" [2], etc.

Therefore we adopt the rule: basic semantics of elements of MDSD modeling should be differentiation oriented and be given in the context of the semantics of other elements. Hence, an automation oriented ontological framework to formally specify and possibly analyze the development processes is desirable.

3 Ontology Creation and Evaluation Framework

3.1 Framework Criteria

Besides that the ontology framework is designed for automations purpose. In defining this ontology framework, we try to reach two most difficult to achieve goals:

(1) Avoid circled semantic definitions. This is guided by requiring the concepts used to give semantics of latter new introduced concepts be limited to the former well defined and structured concept set.

(2) Be deductive to support the coherent semantic transition from the ontology framework to specific implementation. This is guided by the introduction of a serial of concepts differentiation and classification modes.

3.2 Ontology Creation

Concept: Human brain representations for real or virtual things.

Element: The start point of cognition thoughts, element is the first concept. This is the same as in many modeling languages including UML.

Notation: It represents two aspects of meanings of things, the existence and identification. Existence meaning acknowledges the thing or concept as an element. It further differs in the existence time. The identification [25] meaning acknowledges the importance of the meaning of the uniqueness of the element. Both existence meaning and identification meaning should be differed as independent or dependent on other elements.

We only talk about concepts or elements in the MDSD modeling which can be labeled as notations.

Notation differentiation mode: Variation patterns of the focus of the two meanings define the elementary semantics which helps to differentiate with each other.

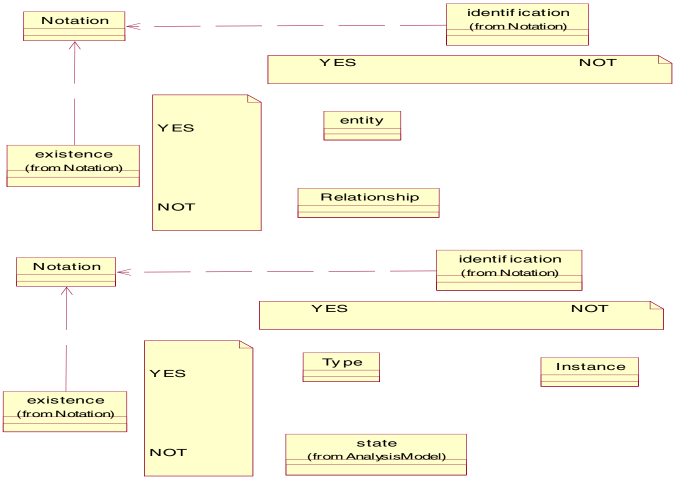


Fig. 2. Notation differentiation mode

Usage: Although the differentiation mode only can differentiation four patterns at a time, by reclusively using this pattern on meaningful element set, many concepts can be differentiated. We have used this mode to help differentiating and giving semantics for elements of our ontological framework. Upper of Fig. 2 shows the differentiation among the entity concept and the relationship concept. Entity concepts emphasis both on identification meaning and existence meaning of notations compared with relationship concepts. The existence of relationship concepts always depend on the existence of related entities. The meaning of relationships is only on the identification. Lower of Fig. 2 show the differentiation among {type, instance, state}. Compared with each other, the identification meaning of an instance concept depends on the corresponding type concept. The state concept is used only to give identification of the thing or things of certain time, not deliberate on showing the existence meaning.

3.3 Framework Overview

Fig. 3 shows part of our ontological framework. We choose the ER model as the start. The MDSD modeling is different from traditional static data modeling. It covers not only static aspect but also dynamic aspect. The ER model is limited to the static area modeling. It lacks necessary explicit representations for runtime entities, and relationships of dynamic runtime or time relevant aspects of MDSD modelling. To improve the situation, we explicitly add the runtime concept instance of OOP to entity set of ER model. The extended ER model owns an entity set {type, attribute, instance}, and is called Runtime Entity-Relationship (RER) [9] model. Therefore the whole element set of ER model "{{entity}, {relationship}}" is extended from "{{type, attribute}, {relationship}}"to "{{type, attribute, instance}, {relationship}}". It implicitly also extends the scope of relationship expression from static data modeling to dynamic behavior modeling. Then RER model supports both the area of static data modeling and the area of dynamic behavior modeling. Please refer to [9] for details of the extension.

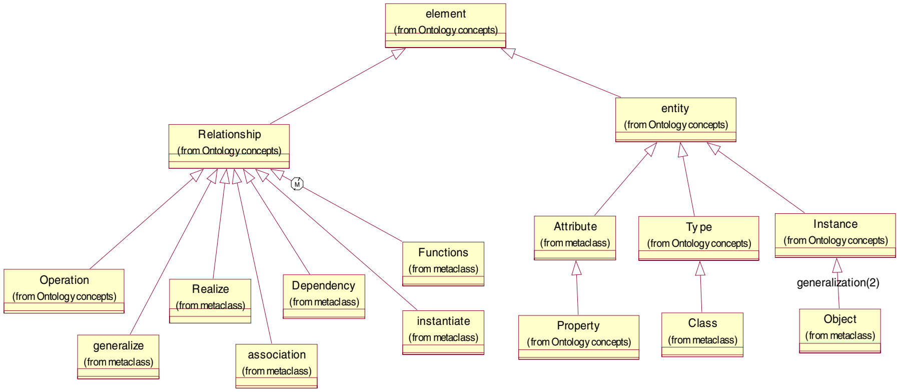


Fig. 3. Part of the ontology mainframe

The attribute means only the function result entity of the function meaning attribute of the ER model. Therefore it is selected to represent the concepts of property, slot, etc. Attribute are considered as a first class entity in the same manner as type concepts in this framework. It is because that they are in fact interdependent. The implementation of the identification meaning of a type is indispensable on the existence of attribute types. Here attribute types and attribute values should be differenced. The identification of an instance depends on the attribute values.

The relationship as a fundamental concept is also the alias of many concepts including function, operation, association, etc. Some proofs can be found from the philosophical understandings, e.g., “An operation is the expression of a relation between the structures of its result and of its bases” [6]. These replacements will surely imply the loss of some individual semantics but the gains of understanding improvement in the whole MDSD modeling far overrun the loss. The relationship should only connect entity elements. No relationship is allowed among relationships or relationships and entities. But the relationships to entities (weak entities) transformation will compliment the expression requirements.

4 Implementations

4.1 Supports in MDSD Modeling

Inspired by Leon J. Osterweil’s “software processes are software too” [3], we start the investigation by creating an ideal model for MDSD in mind which is a coherent and consistent programmable model. Therefore information transferring and transformation are coherent from MDSD model products to the ultimate software products. Fig. 4 shows the architecture of our solution.

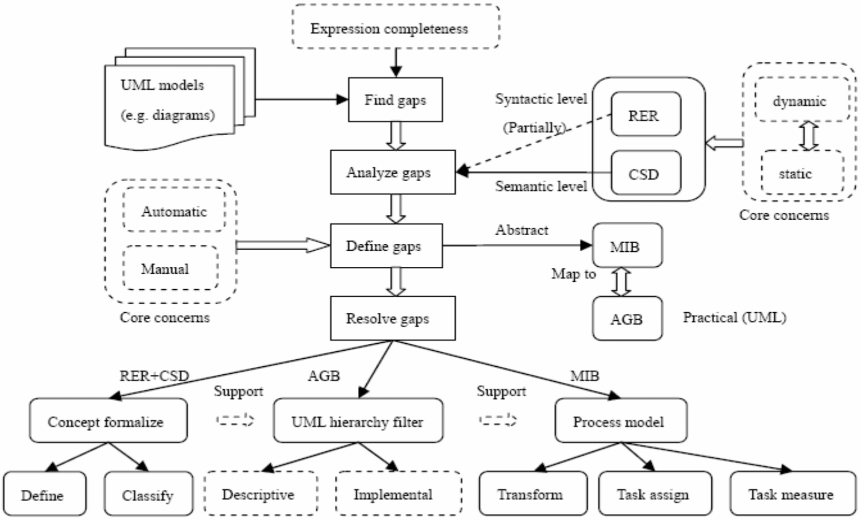


Fig. 4. Solution architecture

First by analysing the UML models from the viewpoint of Section 2.1, expression deficiencies e.g. gaps, are discerned. Then the modeling mechanism is further abstracted into a framework called MIB in which “manual vs. automatic” discussion is the core. This is based on philosophical hypotheses on differentiation basic concepts and context model of discussion MDSD processes which is called CSD [16].

The CSD can function as a concept definition or formalization framework. Using it, expression gaps can be analyzed at semantic level. After that, gaps can be differentiated and defined using the MIB. To make the approach more directly applicable for UML, MIB is replaced with a UML correspondence called AGB [16]. Empirical applications include concept formalization, UML hierarchy classification, and model transformation. This approach is also designed for improving automation task definition, task assign and task measure.

4.2 Initial Results

(1) Transformation rules generation

By introducing the RER model as the description language of MDSD model, all observable changes in MDSD models can be described within the entity set and relationship set of the RER model. The procedure generally involves two main steps:

1) Entity mapping: map relevant composing elements of a concept to the entity set of RER model. Further differentiation from the cardinality aspect.

2) Relationship mapping: currently this is implemented with a relationship metamodel [16] filtered from the UML architecture. For details of the generation algorithm please refer to [16].

(2) UML architecture optimization

By using the RER model as lightweight ontology for modeling MDSD conceptually. Different layers of UML metamodel hierarchy in the context of modeling MDSD are analyzed. Some promising modifications [9] of UML architecture for modeling MDSD completely are suggested.

4.3 Other Applications

Software design patterns and software architectures have greatly benefited the state of art modeling techniques. In the future, we plan to extend our investigation scope from currently analyzing individual entities and relationships to more complex structures of design patterns and software architectures. Some elementary proofs have already been found which may help to relate two most important concepts of relationship and state, e.g., “Relationship: The condition or fact of being related; connection or association [15]”, “State: A condition of being in a stage or form, as of structure, growth, or development [14].” The same locations in the upper and lower parts in Fig. 2 of the two concepts comply with the unified attribution to “condition” kind of the generally selected explanations. This connection will greatly improve the feasibility of our approach to the areas of component oriented developments, state machines and also messages oriented communication facilities.

5 Related Work

The Ontology Definition Metamodel (ODM) [13] provides metamodel for ontology definition supporting ontology development. It uses a collection mode to cover many existing ontology, such as RDF, SCL, DL and OWL. W3C have started a working group to explore best practices and design patterns for OWL. However, this group focuses on ontology construction and does not help with more general issues on MDSD, automation, evaluation, etc. Our framework using a comparision mode which will relate the relevent concepts in different languages. We also give deductive ontology creation modes and steps to support lifting metamodel descriptions (e.g. of domain specific languages of MDA) to ontologies which is not the focus of most organizations including ODM and W3C.

There are some similar other unification efforts such as those of the pUML group. They always face the biggest obstacle of the unwillingness of business society to accept changes at the assumed huge cost. However our techniques, e.g., filtering instead of modifying, provide some coming over alternations for implementation of this approach.

6 Conclusion and Future Directions

This paper addresses the importance of creating an ontology framework for improving the automation level of MDSD. Complex concern set involved is discussed. Then an automation oriented ontological framework to specify and analyze the development processes is proposed. By providing general precise and consistent semantics for development elements, this framework will considerably improve the development of models of automation oriented development MDSD processes. The strength of the approach lies mainly in providing deductive modes to support ontology creation and evaluation which differs from most currently existing approaches. These development guidelines will aid the realization of coherent transitions of understandings among ontologies and specific implementations.

Elementary early applications and initial results [9], [16], [29] seem very promising. Since this field is rather new, and few people have experience in the development of real-world systems, in the future we will collect more example application scenarios that illustrate common problems and challenges. The phase results of our currently experimental applications on inter-transformations and unifications of semantics of existing modeling languages, e.g., ODM, RDF, SCL, DL and OWL, etc, are also very encouraging.

Acknowledgments

This research was supported in part by grants from 973 Program of Chinese Ministry of Science and Technology (2002CB312103), from National Natural Science Foundation of China (60433030 and 30270466), and from the Chinese Academy of Sciences (0302037). Many thanks to friends of SERA 2005 for valuable early comments.

References

1. Gerd Wagner. The Agent-Object-Relationship metamodel: towards a unified view of state and behavior. Inf. Syst. 28(5): 475-504 (2003).
2. Brian Henderson-Sellers. UML - the Good, the Bad or the Ugly? Perspectives from a panel of experts. Software and System Modeling 4(1): 4-13 (2005)

3. Leon J. Osterweil. Understanding process and the quest for deeper questions in software engineering research. ESEC / SIGSOFT FSE 2003: 6-14
4. Alexander Egyed. Heterogeneous Views Integration and its Automation, Ph.D. Thesis, Univ. of Southern California, 2000.
5. Kendall Scott. UML Explained. Publisher:Addison-Wesley , 2001. ISBN: 0 201 72182.
6. Ludwig Wittgenstein. Tractatus Logico-Philosophicus. 1918. http://www.wmelchior.com/wis/philo/wittgenstein/works/tractatus/tlp.htm
7. Natasha Noy,Alan RectorDefining N-ary Relations on the Semantic Web: Use With Individuals.W3C Working Draft 21 July 2004.This version//www.w3.org/TR/2004/WD-swbp-n-aryRelations-20040721
8. Barry W. Boehm, Daniel Port. Conceptual Modeling Challenges for Model-Based Architecting and Software Engineering (MBASE). Conceptual Modeling 1997: 24-43
9. Yucong Duan, Xiaolan Fu, S.C. Cheung, Yuqing Gu. An Entity-Relationship Model Based Conceptual Framework for Model Driven Development, In Proc. of IASTED Int'l Conf. on Software Engineering (SE2006), Innsbruck, Austria, Feb. 14-16, 2006, pp. 200-205.
10. Peter P. Chen: The Entity-Relationship Model - Toward a Unified View of Data. ACM Trans. Database Syst. 1(1): 9-36 (1976)
11. B.Tekinerdogan. Synthesis Based Software Architecture Design, Ph.D. thesis, University of Twente, The Netherlands, March 2000.
12. Ross Jeffery. Achieving Software Development Performance Improvement Through Process Change. Invited talk in Software Process Workshop 2005, Beijing, China, May26, 2005. http://www.cnsqa.com/cnsqa/jsp/html/spw/ppt/Achieving%20Software%20Development%20Performance%20Improvement.pdf
13. DSTC, Gentleware, IBM, Sandpiper Software.Ontology Definition MetaModel. August 23rd, 2004. http://codip.grci.com/odm/draft/submission_text/ODMPrelimSubAug04R1. pdf.
14. M. Dahchour and A. Pirotte. The semantics of reifying relationships as classes. In Proc. of the 4th Int. Conf. on Enterprise Information Systems, ICEIS'02, pages 580-586, Ciudad Real, Spain, April 2002.
15. The American Heritage® Dictionary of the English Language, Fourth Edition. Copyright © 2000 by Houghton Mifflin Company.
16. Yucong Duan, S.C. Cheung, Xiaolan Fu and Yuqing Gu, A Metamodel Based Model Transformation Approach, in Proc. of SERA 2005, MI, USA, Aug. 11-12, 2005, pp. 184-191.
17. Natalya Fridman Noy, Deborah L. McGuinness. Ontology Development 101: A Guide to Creating Your First Ontology . Stanford Knowledge Systems Laboratory Technical Report KSL-01-05. March 2001.
18. Guarino, N. and Welty, C. Evaluating Ontological Decisions with OntoClean, Communications of the ACM, 45(2): pp. 61~65
19. Liwu Li, Ontological modeling for software application development, Advances in Engineering Software, v.36 n.3, p.147-157, March 2005
20. Gruninger, M. and Fox, M.S.. Methodology for the Design and Evaluation of Ontologies. In: Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing, IJCAI-95, Montreal. (1995)
21. McGuinness, D.L., Fikes, R., Rice, J. and Wilder, S.. An Environment for Merging and Testing Large Ontologies. Principles of Knowledge Representation and Reasoning: Proceedings of the Seventh International Conference (KR2000). A. G. Cohn, F. Giunchiglia and B. Selman, editors. San Francisco, CA, Morgan Kaufmann Publishers. (2000)
22. Rosch, E.. Principles of Categorization. Cognition and Categorization. R. E. and B. B. Lloyd, editors. Hillside, NJ, Lawrence Erlbaum Publishers: 27-48. (1978)

23. Uschold, M. and Gruninger, M. (1996). Ontologies: Principles, Methods and Applications. Knowledge Engineering Review 11(2).
24. Quine, W.V.O. Ontological Relativity and Other Essays. Columbia University Press, New York, London, 1969.
25. Guarino, N. and Welty, C. Identity, unity, and individuality: Towards a formal toolkit for ontological analysis. In Proceedings of ECAI-2000: The European Conference on Artificial Intelligence. IOS Press, Berlin, Germany, 2000.
26. York Sure, Asunción Gómez-Pérez, Walter Daelemans, Marie-Laure Reinberger, Nicola Guarino, Natalya Fridman Noy: Why Evaluate Ontology Technologies? Because It Works!. IEEE Intelligent Systems 19(4): 74-81 (2004)
27. Rich Hilliard. Aspects, Concerns, Subjects, Views, ... In First. Workshop on Multi-Dimensional Separation of Concerns in. Object-oriented Systems (at OOPSLA ’99), 1999.
28. Holger Knublauch. Ontology-Driven Software Development in the Context of the Semantic Web: An Example Scenario with Protégé/OWL. International Workshop on the Model-Driven Semantic Web, Monterey, CA (2004)
29. Yucong Duan, Yuqing Gu, Xiaolan Fu. A Conceptual Approach to Modeling Model Driven Development Processes, in Proceedings of International Conference on Systems (ICONS'06), Mauritius, April 27-29, 2006, IEEE Computer Society Press. (in press)
30. Roger Lee, Ashok Harikumar, Chia-Chu Chiang, Hae Sool Yang, Haeng-Kon Kim, Byeongdo Kang: A Framework for Dynamically Converting Components to Web Services. SERA 2005: 431-437

An AHP-Based Evaluation Model for Service Composition

Xiaoqin Xie[1] and Kaiyun Chen[2]

[1] Computer Science and Technology Institute, Harbin Engineering University,
150001 Harbin, China
xiexiaoqin@tsinghua.org.cn
[2] Mechanics and Electronics Engineering Institute, Harbin Engineering University,
150001 Harbin, China
chenkaiyun@tsinghua.org.cn

Abstract. In this paper we present the Analytic Hierarchy Process (AHP)-based composition evaluation model (ASCEM) for service composition, which provides quantitative and global evaluation for selecting composition plans. Quantitative factors and global computing formula are defined also. Thus ASCEM enables dynamic service composition. ASCEM features in following. Firstly, AHP enable a more objective weight-allocating for evaluation factors and the hierarchical model provides more scalability. Secondly, the evaluation parameters include not only the quality properties but also the properties of reasonability and granularity for business processes. Thirdly, this model gives a quantitative and global result. A file-workflow process example is taken to illustrate how to use ASCEM and the results prove that the model is feasible and correct.

1 Introduction

Composition-based software development is one important method for software reuse. The development procedure includes such steps as service discovery, selection, adaptation, composition, and execution [1]. Evaluation for the service or composition result is critical for the success of the selection, composition and replacement steps. However, current researches mostly either focus on evaluating the single service instead of composition or lack the quantitative global criteria. The qualitative approaches can only tell the service consumer that whether the service can be used or not. How to rank or rate the candidate services for this approach is difficult. These impede the feasibility of automatic service composition.

The process-based composition of Web services, especially dynamic process-driven service composition, is gaining a considerable momentum as an approach for the effective integration of distributed, heterogeneous, and autonomous applications [2]. But, dynamic service composition presents significant challenges and requires addressing a number of critical issues such as discovering and identifying relevant services, formulating and ranking (and selecting) composition plans using current context, goals, constraints and costs and so on. Planning technologies is exploited for

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 756 – 766, 2006.

automatic service composition in many researches such as [3][4]. Composition plan describes how several component services are aggregated into one composite service. It is created at runtime based on dynamically defined composition objectives, their semantic descriptions, constraints, and available services and resources[5]. To assure that the whole composition succeeds, how to evaluate the candidate composition plans and select the optimal plan is critical. Furthermore, dynamic composition demands for generating and selecting plans automatically. So this paper focuses on the research of quantitative evaluation for composition plans. This is also the motivation of our research.

Many approaches have been put forward to address the issue about the runtime selection of component services for a composite service [6]. However, previous approaches in this area have not identified these criteria about business process that plays an important role in service composition. For example, LJ Zhang gives an evaluation method that focuses on selecting one component for an individual task instead of the composite service [7]. L.Zeng et al. [6] proposed a global quality-driven evaluation model. But they ignore the reasonability and granularity factors for business processes. In addition, many evaluation approaches are not able to give a quantitative result for composition plan.

In this paper, we present Analytic Hierarchy Process (AHP)-based synthesized composition evaluation model (named as ASCEM) for service composition. ASCEM enables dynamic service composition. The salient features of our model are: 1) AHP-based model; 2) Quantitative and global evaluation; global evaluation means that: not only the properties of the simple component but also other composition properties such as granularity and number of composite component and so on are considered in this model. 3) Process-driven evaluation.

The rest of the paper is organized as follows. Section 2 describes the evaluation problem. Section 3 defines some key concepts throughout the paper and presents the ASCEM model and its building-up steps. Section 4 presents a file-process workflow example to illustrate ASCEM. Finally, Section 5 discusses related works and gives the compare with other evaluation methods, and Section 6 draws some conclusions.

2 The Problem of Service Composition Evaluation

The composition-based software development involves five steps in order: retrieval, selecting, adapting, composing and executing. The composition plan is generated before composing and executing steps and integrates the result in the forgoing three steps. In this paper, the meaning of the evaluation for service composition is the same as the evaluation for composition plan.

The composition plan can be defined as follows:

Definition 1. Composition plan (*CP*) is a path:

$$CP=\mathrm{v}_1 \rightarrow \mathrm{v}_2 \rightarrow \ldots \rightarrow \mathrm{v}_\mathrm{n},\ \mathrm{v}_\mathrm{i}= (cr_i, s_i),\ 1 \leq i \leq \mathrm{n}$$

Where $(cr_i, s_i) \in CR \times S$, CR is composition request and S is the set of component services or composite services. In a composition plan, there are two kinds of services. The first is component service, which can be used as soon as they are taken out of repository. The other is composite service that includes multiple services. One composite service is composed of component services. When a composite service is included in another composite service, the first composite service is called also a component service. As for the composition relation, the composition template and stored template concepts describe them, which can be referred to Section 3.

The evaluation problem of composition plan can be defined as follows:

Assume that $CPS=\{CP_1,CP_2,\ldots,CP_n\}$ and CP_i is the candidate composition plan. And assume that the evaluation formula is $f()$. Thus the goal of the evaluation is to select one plan CP_i such that the value of $f(CP_i)$ is maximal.

3 Analytic Hierarchy Process-Based Synthesized Composition Evaluation Model (ASCEM)

In order to get a quantitative and global evaluation result for composition plans, we will employ the theory of Analytic Hierarchy Process for the composition plan evaluation. This section illustrates the evaluation model and its building-up steps.

3.1 The Analytic Hierarchy Process Method

The Analytic Hierarchy Process (AHP) is one qualitative and quantitative method of the analysis on multi-objective decision. AHP decompose a complicated problem into component elements, which are organized into a hierarchical structure in terms of the relations among component elements. The pair-wise comparison of component elements determines the relative decision element weights. Thus an order on decision weights of component elements can be worked out. The basic steps of AHP are as following:

1. Analyze the problem, and determine the AHP factors. In addition, allocate factors to different level of hierarchy model.
2. Build a pare-wise comparison matrix for AHP factors in terms of the relative importance between two factors.
3. Check consistence of the matrix.
4. Compute the eigenvector and eigenvalue of the matrix. The best weights fit into the pair-wise comparisons matrix.

3.2 Preliminary Evaluation Factors for Composition Plan

The evaluation factors for composition plan involve service type, number and complexity. The number and the complexity of component services that are included in a composite service determine the performance and efficiency of the composite service. So the preliminary evaluation factors come from the following three aspects:

- Difference between retrieved and ideal component service.
- Difference between retrieved and ideal composition relation.
- Complexity of composite service.

These three aspects determine the evaluation factors in ASCEM. When each component service is retrieved out of the service repository, it is annotated a similarity property by search engine, which is called Similarity of Service (*SS*). *SS* belongs to [0,1]. About how to calculate the similarity, we will describe it in another paper.

Definition 2. (Synthesized Similarity of Compositive Service, SSCS) SSCS is a weighted-sum of *SS*, which is as following:

$$SSCS = \sum_{i=1}^{N} w_i SS_i \tag{1}$$

Where: *N* means the number of component services included in composition plan, SS_{i} means the similarity of ith service, w_{i} means the relative importance $I(s_i)$ between two component services. $I(S_i)$ can be calculated as following:

$$I(s_i) = \alpha * default(s_i) + \beta * \frac{num(s_i)}{d} \tag{2}$$

Where: $default(s_i)$ can be set by domain specialist, *d* means the times the repository was accessed. $num(s_i)$ means the times the service in repository was selected. αandβ are controlled parameters which can be set by domain specialist or be probability value. Theirs value domains belong to [0,1].

The definition of second evaluation factor-Similarity of Business Process, which is based on the notions of stored and composition templates which were defined by Brahim M firstly[8] is as following:

Definition 3. (Stored Template) A stored template, which is defined by end users or achieved through search engine, describes the composition relations among component services. It is defined as a directed graph-Stored Template Graph STG(*V*,*E*), where *V* is a set of component services included in a composite service and *E* is a set of edges. *V* can be presented as $\{v_i\}$ where *i* belongs to [1,*N*] and *N* means the number of component services. An edge (v_i, v_j) belongs to *E* if there exists composition relation between v_i and v_j. Stored templates are saved in template repository (TL).

Definition 4. (Composition Template) A composition template which is defined or confirmed by domain specialists is build for each composite service(*CS*) and describes the *CS*'s general structure[8]. It is modeled by a directed graph-Composition Template Graph CTG(*V*, *E*), where the V and E definitions are similar to Definition 2.

The difference between stored and composition templates lies in that the composition templates are achieved from a composition relation graph related to a stored template instead of being saved in template repository.

In fact, a composite service always reflects a real business process. The similarity between CTG and STG reflects whether the retrieved composite plan agrees to the business requirement. Thus the problem is converted to a graph similarity problem. According to different relationships between two graphs, following gives the similarity calculation under four conditions.

Definition 5. (Similarity of Business Process, SBP) SBP describes the similarity between CTG and STG. It can be modeled as following:

$$SBP = \begin{cases} 1 & \text{if } \exists \mathrm{STG} \in TL \cap \mathrm{CTG} = \mathrm{STG} \\ \log 6 & \text{if } \exists STG \in TL \cap \mathrm{CTG} \supset \mathrm{STG} \\ \log 3 & \text{if } \exists \mathrm{STG} \in TL \cap \mathrm{CTG} \subset \mathrm{STG} \\ 0 & \text{if} \neg \exists STG \in TL \cap (\mathrm{CTG} = \mathrm{STG} \cup \mathrm{CTG} \subset \mathrm{STG} \cup \mathrm{CTG} \supset \mathrm{STG}) \end{cases} \tag{3}$$

CTG is on behalf of the business process. The algorithm for checking the condition in above equation is through comparing STG and CTG matrices.

A composition plan may involve multiple composite services, namely multiple business processes. So the following will define the second preliminary evaluation factor to describe the process property of composite services.

Definition 6. (Synthesized Similarity of Business Process, SSBP) SSBP is the weighted-sum of process similarity of composition template, which is presented as following:

$$SSBP = \sum_{i=1}^{M} w_i SBP_i \tag{4}$$

Where: M means the number of composite services. SBP_{i} means the business process similarity. w_{i} means the importance of one composition template in composition plan. w_{i} can be worked out by calculating the eigenvalue of pair-wise comparison matrix in terms of relative importance $I(ct_i)$ between two composition templates. $I(ct_i)$ can be calculated as following:

$$I_{bp}(\mathrm{ct}_i) = \lambda * default(ct_i) + \gamma * \frac{e}{f} \tag{5}$$

Where: ct_{i} means the composition template. The $default(ct_{\mathrm{i}})$ is setup by domain specialists. The f means the times the service repository is accessed. The e means the times the ct_{i} is selected which can be gotten by counting the times that the composition rules are triggered. λ and ɣ are controlled parameters which can be set by domain specialist or be probability value. Theirs value domains belong to [0,1].

To determine how many component services included in a composite service is proper is a difficult and complex problem. We call the property that describes the complexity of component services as the granularity of service. YX Wang proposed the idea of allocating the weight to software control structure and defined the concept of software complexity[9]. Based on his weight-allocating solution, combing the feature of process-oriented service composition, we give the definition of the third evaluation factor in the following.

Definition 7. (Granularity of Composite Service, GCS) GCS is a measurement unit for describing the complexity of composite service. It is modeled as following:

$$GCS(s) = w_I N_I + w_O N_O + w_C N_C + w_{OP} \sum_{i=1}^{n} WOP_i \tag{6}$$

Where: s means the composite service. N_I and N_O mean the number of input and output parameter of the composite service respectively. N_C means the number of component services involved in s. WOP_i means weight allocated to composition operator which describe the concrete composition such as sequential composition, parallel composition, iterative composition, selective composition and so on. w_I , w_O , w_C , w_{OP}

is the eigenvalue of pair-wise comparison matrix which is constructed in terms of the relative importance of N_I , N_O , N_C and *WOP*.

The *GCS* is negative, i.e., the higher the value is, the lower the quality is. The *SSBP* and *SSCS* are positive criteria, i.e., the higher the value is, the higher the quality is. For negative *GCS*, values are scaled according to the following equation.

$$process(GCS) = \frac{GCS_{\max} - GCS}{GCS_{\max} - GCS_{\min}} \tag{7}$$

Where: GCS_{max} and GCS_{min} mean the maximum and minimum value of service granularity respectively.

3.3 ASCEM Model

According to AHP principles, figure 1 depicts the ASCEM model for evaluating composition plan. The model building-up steps are as following:

Step 1. Build up the hierarchical structure.

As depicted in figure 1, the top level is the synthetical evaluation result expressed as *S*. *S* is determined by such three criterions as *SSCS*, *SSBP* and *GCS*. A composition plan includes *N* component services, so *SSCS* is settled upon SS_1, SS_2, ..., SS_n. A composition plan may include *M* composition templates each of which corresponds to one business process. So *SSBP* is confirmed by the sub criterions of SBP_1 , SBP_2 , ..., SBP_M. In addition, *GCS* is determined by the four sub-criterion of N_I , N_O , N_C and the sum of *WOP*.

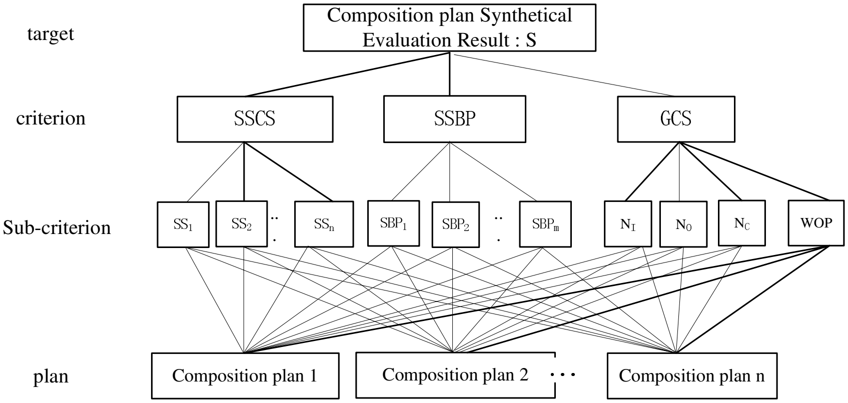


Fig. 1. AHP-based Synthesized Composition Plan Evaluation Model (ASCEM)

Step 2. Taking aim at *SSCS*, calculate the weights of SS_1, SS_2, ..., SS_N. After constructing the pair-wise comparison matrix of SS_1,*to* SS_N, calculate the maximum eigenvector of the matrix and normalize it, the weight of SS_1 ,*to* SS_n $(w_{SS_1}, w_{SS_2}, ..., w_{SS_N})$ can be achieved.

Step 3. Taking aim at *SSBP*, calculate the weights of SBP_1 , SBP_2 , ..., SBP_M. By using the same calculating method as Step 2, the weight of SBP_1 to SBP_M $(w_{SBP_1}, w_{SBP_2}, ..., w_{SBP_M})$ can be achieved.

Step 4. Taking aim at *GCS*, calculate the weights of N_I, N_O, N_C and *WOP*. By using the same calculating method as Step 2, the weight of N_I, N_O, N_C and *WOP* $(w_{N_I}, w_{N_O}, w_{N_C}, w_{WOP})$ can be achieved.

Step 5. Taking aim at *S*, calculate the weights of *SSCS*, *SSBP* and *GCS*. After constructing the pair-wise comparison matrix of *SSCS, SSBP* and *GCS,* weight vector $(w_{SSCS}, w_{SSBP}, w_{GCS})$ is achieved by normalizing the eigenvector.

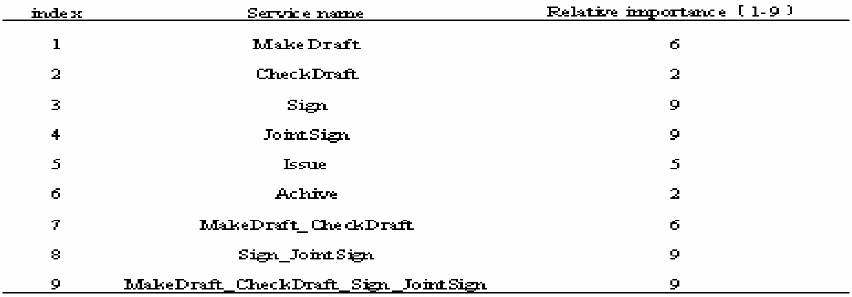
Step 6. Compute the synthetical evaluation result for composition plan. The computing formula is as following:

$$\begin{aligned} S = {} & w_{SSCS} \times (w_{SS_1} \times SS_1 + w_{SS_2} \times SS_2 + \ldots + w_{SS_N} \times SS_N) \\ & + w_{SSBP} \times (w_{SBP_1} \times SBP_1 + w_{SBP_2} \times SBP_2 + \ldots + w_{SBP_M} \times SBP_M) \\ & + w_{GCS} \times process((w_{N_I} \times N_I + w_{N_O} \times N_O + w_{N_C} \times N_C + w_{WOP} \times WOP)) \end{aligned} \tag{8}$$

4 The Example

This section illustrates how to use ASCEM model through a file-flow process example. We assume that the service composer demand for a “file-sending process” service. But there are no directly matched services in service repository. Currently there exist such service implements as depicted in table 1. Search engine has retrieved three candidate composition plans. In the following, we will use ASCEM model to evaluate the three composition plans and select one. The relative importance of factors adopts the 9-point scale, in which 1 means equally importance, 3 means moderately more importance, 5 means strongly more importance, 7 means very strongly more importance and 9 means extremely more importance. The values of relative importance depicted in the right column of table 1 and are given by domain specialists. In addition, there exist two different stored templatesin template repository.

Table 1. The Service Implement in Service Repository



index	Service name	Relative importance [1-9]
1	MakeDraft	6
2	CheckDraft	2
3	Sign	9
4	JointSign	9
5	Issue	5
6	Achive	2
7	MakeDraft_CheckDraft	6
8	Sign_JointSign	9
9	MakeDraft_CheckDraft_Sign_JointSign	9

In accordance to different requirement and application contexts, three composition plans returned by search engine are described in table 2. For simplicity, we assume that there exists only one service for one task.

Table 2. Three Composition Plans

index	Selected service	Related stored template
Composition plan1	MakeDraft,CheckDraft,Sign,JointSign,Issue,Archive	Template 1, Template 2
Composition plan2	Sign,Archive,MakeDraft_CheckDraft,Sign_JointSign	Template 2
Composition plan3	MakeDraft_CheckDraft_Sign_JointSign,Issue,Archive	Template 2

In a general way, it is more difficult to modify a composite service than replacing a simple component. Furthermore, we prefer a simple service-based system to a complex one. So the scales of importance of *SSCS*, *SSBP* and *GCS* factors are 8, 9 and 5 respectively. The pair-wise comparison matrix A_S is as following:

$$A_S = \begin{bmatrix} 8/8 & 8/9 & 8/5 \\ 9/8 & 9/9 & 9/5 \\ 5/8 & 5/9 & 5/5 \end{bmatrix} = \begin{bmatrix} 1 & 8/9 & 8/5 \\ 9/8 & 1 & 9/5 \\ 5/8 & 5/9 & 1 \end{bmatrix}$$

By using the function eig() of MATLAB6.5, the maximum eigenvalue $\lambda_{\max}$ is achieved as 3. And the Satty consistency checking parameter $C.I = \frac{\lambda_{\max} - n}{n-1}$ equals to zero. So the matrix A_S is consistent absolutely. The eigenvector is (0.6136,0.6903,0.3835). So the normalized factor weight vector $(w_{SSCS}, w_{SSBP}, w_{GCS})$ equals to (0.6136,0.6903,0.3835). The relative importance weights of N_I, N_O, N_C and *WOP* are 5,5,9,9 respectively. Thus (w_I, w_O, w_C, w_{OP}) equals to (0.3434,0.3434, 0.6181,0.6181). We assume that the composite service with minimum granularity has only one input, one output and includes no component service, so according to equation (6), $GCS_{\min}$ equals to 1.3049. In the same way, we assume that the composite service with maximum granularity has three input, three output and includes fifty component service, and the sum of WOP equals to fifteen that means each composition operator occurrence once and only once. So the $GCS_{\max}$ equals to 42.2369. Based on above setup, the following will give the detailed computation on ASCEM for evaluating three different composition plans.

- Composition plan 1

This solution has six component services that are *MakeDraft*,*CheckDraft, Sign, Joint-Sign*, *Issue* and *Archive* respectively. The corresponding service similarities are 0.8, 0.7, 0.9, 0.7, 0.8 and 0.4.

According to table 1, one pair-wise comparison matrix is gotten which is presented as: $\begin{bmatrix} 6/6 & 6/2 & 6/9 & 6/9 & 6/5 & 6/2 \\ 2/6 & 2/2 & 2/9 & 2/9 & 2/5 & 2/2 \\ 9/6 & 9/2 & 9/9 & 9/9 & 9/5 & 9/2 \\ 9/6 & 9/2 & 9/9 & 9/9 & 9/5 & 9/2 \\ 5/6 & 5/2 & 5/9 & 5/9 & 5/5 & 5/2 \\ 2/6 & 2/2 & 2/9 & 2/9 & 2/5 & 2/2 \end{bmatrix}$, and the eigenvalue $\lambda_{\max}$ equals to 6. So the normalized weight vector (w_1,w_2,w_3,w_4,w_5,w_6) equals to (0.3948, 0.1316, 0.5922, 0.5922, 0.3290, 0.1316), there holds:

1) *SSCS*=w_1*0.8+w_2*0.7+w_3*0.9+w_4*0.7+w_5*0.8+w_6*0.4=1.6713.

2) Because the composition template fits to the second stored template very well, *SSBP* equals to 1.

3) *GCS*=0.3434+0.3434+0.6181*6+0.6181*(1+1+4)=7.4859.
Process(*GCS*)=(42.2369-7.4859)/40.932=0.849.

To sum up, the synthetical evaluation result S_1 for composition plan 1 is worked out as: S_1=w_{ssbc}*1.6713+w_{ssbp}*1+w_{gbc}*0.849=2.0414.

- Composition plan 2

This solution has four component services that are *MakeDraft_CheckDraft, Sign_ JointSign*, *Issue* and *Archive* respectively. The corresponding service similarities are 0.75, 0.8, 0.8 and 0.4 respectively.

By using the same calculate method as depicted in composition plan 1, we get the following result: 1) *SSCS*=1.3655; 2) SSBP equals to log3, namely 0.4771;3) *GCS*=3.7773. Furthermore, Process(*GCS*)=(42.2369-3.7773)/40.932=0.9396.To sum up, the synthetical evaluation result S_2 for composition plan 2 is: S_2=1.5275.

- Composition plan 3

This solution has three component services that are *MakeDraft_CheckDraft_Sign_ JointSign*, *Issue* and *Archive* respectively. The corresponding service similarities are 0.75, 0.8 and 0.4 respectively.

By using the same calculate method as depicted in composition plan 1, we get the following result: 1) *SSCS*=1.1012; 2) *SSBP* equals to log3, namely 0.4771;3) *GCS*=3.1592. Furthermore, Process(*GCS*)=(42.2369-3.1592)/40.932=0.9547. To sum up, the synthetical evaluation result S_3 for composition plan 3 is: S_3=1.3711.

According to the above computation result, we can see that S_1 is best and S_3 is worst. That this result agrees to the fact proves that our model is feasible and correct.

5 Related Works

This section briefly discusses difference between our work and other related works.

B. Medjahed introduced a Quality of Composition model to assess the quality of generated composite services[8]. This model defined such properties as *composition soundness*, *composition ranking* and *composition completeness.* Our work is similar to this one. But his model did not give a synthetical evaluation result.

LZ Zeng et al proposed a web service quality model for evaluating basic and composite services [7], which characterize non-functional properties that are: *execution price*, *execution duration, reputation, reliability* and *availability*. The different between our work and theirs lies in that we consider not only the non-function properties but also the semantic of business process, such as the reasonability of business process and the service granularity.

LJ Zhang et al. considered the business process composition as a set of services[7]. They defined risk-minimum function to describe the difference between generated composition plan and original request. Executing time and price are taken as the evaluation criterion. Besides their goal to choose one service from multiple service providers, our method provides the function for choosing optimal composition plan for composing component services.

In addition, in workflow area, there are many researches about QoS. For example, J Cardoso built up a quality model from the time, cost, reliability and fidelity dimensions [10]. But he focused on analyzing, predicting and monitoring the QoS of workflow process, and ignored the dynamic composition of services [6].

Comparing with the above methods, ASCEM characterizes in the following:

1) Based on AHP. So the weights for evaluation factors are more objective and the hierarchical model enabled more scalability.

2) Quantitative and synthetical evaluation for composition plan. In addition, as for the preliminary evaluation factors, Equation 2 and 5 in Section 3 concern not only the users' subjective demands but also the statistical information from the using history.

3) Process-oriented evaluation. ASCEM considers not only the quality properties of services but also the reasonability and granularity of business process.

6 Conclusion

Dynamic selection of component service and composition plan is an important issue in web services composition. In this paper, we have presented a synthetical and quantitative evaluation model ASCEM to evaluate composition plan. By adopting the AHP method to construct the model, ASCEM enables more objectivity and scalability. ASCEM provides an approach to selecting composition plan or composite service and enables dynamic service selection and composition.

We have taken a file-flow process as an example to illustrate how to build up the ASCEM model and compute the evaluation factors. The results show that the proposed approach is feasible and correct. Our ongoing research includes developing more proper criterion for business process composition. And we will apply ASCEM to more real applications to test it.

References

[1] Hafedh Mili, Ali Mili, Sherif Yacoub,Edward Addy. Reuse-Based Software Engineering: Techniques, Organization, and Controls, Publishing House of Electronics Industry, Beijing (English Version), (2003)

[2] B. Benatallah and F. Casati, editors. Distributed and Parallel Database, Special issue on Web Services. Springer-Verlag, (2002)

[3] Dan Wu,Bijan Parsia,Evren Sirin,James Hendler,and Dana Nau. Automatic Web Services Composition Using SHOP2. In Proceedings of 2nd International Semantic Web Conference(ISWC2003),Sanibel Island,Florida,October (2003)

[4] Joachim Peer, Towards Automatic Web Service Composition using AI Planning Techniques(first draft).August 10, (2003)

[5] Manish Agarwal; Manish Parashar. Enabling autonomic compositions in grid environments. Grid Computing, Proceedings. Fourth International Workshop on ,17 Nov. (2003) 34–41

[6] L. Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, and Q. Z.Sheng. Quality Driven Web Services Composition. In Proceedings of the International World Wide Web Conference, Budapest, Hungary, May (2003) 411–421

[7] Liang-Jie Zhang, Bing Li, Tian Chao, Henry Chang. On Demand Web Services-Based Business Process Composition. IEEE (2003) 4057–4064

[8] Brahim Medjahed. Semantic Web Enabled Composition of Web Services:[Phd dissertation in Computer Science and Applications]. Virginia,USA. Virginia Polytechmic Institute and State University. (2004)

[9] Yingxu Wang. Component-Based Software Measurement. (2003) 247–262

[10] J Cardoso. Quality of Service and Semantic Composition of Workflows:[Ph.D. Thesis], USA:University of Georgia, (2002)

Construction of Quality Test and Certification System for Package Software*

Ha-Yong Lee[1], Hae-Sool Yang[2], and Suk-Hyung Hwang[3]

[1] Seoul Univ. of Venture & Information, 1603-54, Seo-cho dong,
Seo-cho gu, Seoul, 137-070, Korea
lhyazby@suv.ac.kr

[2] Graduate School of Venture, HoSeo Univ., 1603-54, Seo-cho dong,
Seo-cho gu, Seoul, 137-070, Korea
hsyang@office.hoseoa.ac.kr

[3] Division of Computer & Information Science, SunMoon University,
100 Kal-san ri, Tang-jung myeon, A-san, Chung-nam,
336-840, Korea
shwang@sunmoon.ac.kr

Abstract. Package software should have the feature that purchasers can discriminate a product suitable for them among a number of software, which belong to the similar kind of product. Purchasers' ability to choose a package software depends on that they can judge whether a package software has the relevant standard conforming through objective quality test process and method or not. For building this system, there are the standards that can be applicable to pack-age software, such as <ISO/IEC 14598-5 : Quality Evaluation Process for Evaluator> and <ISO/IEC 12119 : Information Technology - software package - Quality Requirements & Test>. This study built the system that purchasers can effectively select a pack-age software suitable for their needs, building quality test and certification process for package software and developing Test Metric and application method.

1 Introduction

Due to the rapid spread of personal computers, a variety of package software for personal or office use have been developed, and consequently the liberty of choice has been broadened. Package software should have the feature that purchasers can discriminate a product suitable for them among a number of software, which belong to the similar kind of product. If we want to make a right choice for package software, we should consider whether a package software satisfies the established standard or not through objective quality test process and method.

For building this system, there are the standards that can be applicable to package software, such as <ISO/IEC 14598-5 : Quality Evaluation Process for Evaluator> and

* This research was supported by University IT Research Center Project.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 767 – 776, 2006.

<ISO/IEC 12119 : Information Technology - software package - Quality Requirements & Test>. In case of ISO/IEC 12119, those can use it, such as software developers, organizations for authentication that intend to establish third-party authentication, organizations for approving authentication and test centers, and software purchasers.

This study developed the method that can contribute to quality improvement of package software by building the quality test process for package software based on this standard and developing test metric and application method. This study introduces the present research state related to quality in Chapter 2, and builds the test process for package software from the purchasers' viewpoint in Chapter 3. It introduces quality model for testing package software in Chapter 4, and describes the metric that was developed based on quality model in Chapter 5, and finally describes the conclusion and further studies.

2 Present State of Related Works

2.1 Foreign Trend

Foreign advanced countries in software are continuously trying to establish the standard for quality evaluation. They are on the way to standardize ISO/IEC 9126 as the standard on quality evaluation features and ISO/IEC 14598 as the standard on quality evaluation process. However, it is the actual circumstances that it is very rare they build the specific quality evaluation method and then actually apply it, based on the general contents on standard. And there is a case that they build the practical evaluation system about application, a part of quality features, and then utilize it.

2.2 Domestic Trend

It can be said that now the domestic trend on quality evaluation & test technology has its weak basis on the whole. The related standard for quality evaluation has not been established yet, and the authentication for software's quality system relies on foreign countries, and thereby we can see the basic study in domestic is very weak. Even though domestic software industry regards technology for quality improvement and development of product evaluation technology as the urgent task, it has much difficulty in pushing technology development in itself.

3 Building Certification Process for Package Software

3.1 The Outline of the Process

The outline of the test/certification process is like fig. 1. We constructed it 'certification request and acceptance', 'certification', 'certification announcement and delivery' and 'activity after certification'. The test during the certification makes progress through the self test department or an external test department.

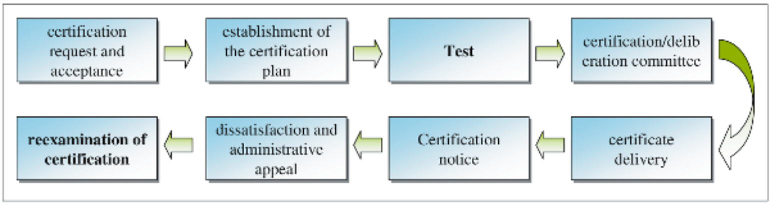

certification request and acceptance
establishment of the certification plan
Test
certification/deliberation committee
certificate delivery
Certification notice
dissatisfaction and administrative appeal
reexamination of certification

Fig. 1. The Outline of the Test/Certification Process

3.2 The Detailed Activity of the Process

3.2.1 Certification Request and Acceptance

The certification applicant presents software quality certification request form, product description, user's manual and software. The certificate authority presents the following items to the certification requester.

- The certification request process, the request method, a fee
- The technical items such as test and certification method
- The effect and the coverage of the certification

The certificate authority can analyze certification request documents and complement the received documents in consultation with the requester if they are unsatisfactory.

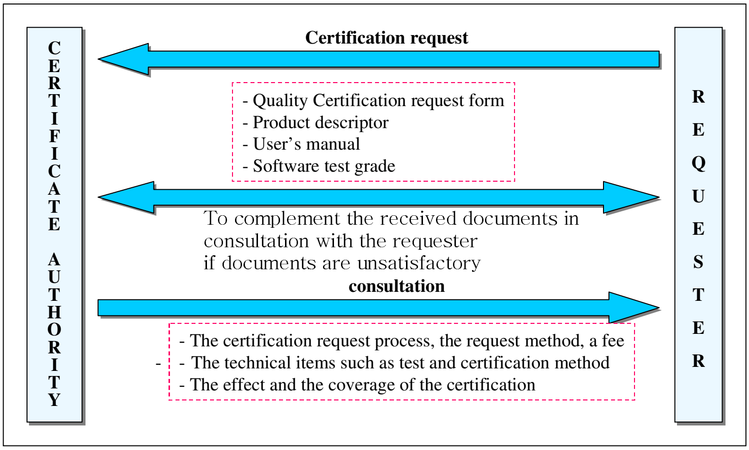

CERTIFICATE AUTHORITY
REQUESTER
Certification request
- Quality Certification request form
- Product descriptor
- User's manual
- Software test grade
To complement the received documents in consultation with the requester if documents are unsatisfactory
consultation
- The certification request process, the request method, a fee
- The technical items such as test and certification method
- The effect and the coverage of the certification

Fig. 2. Certification request and acceptance

3.2.2 Test Asking and Establishment of the Certification Plan

The certificate authority confers the following items with the requester to implement the certification.

- The day's program and the method of the certification
- Required items when the certification is implemented
- Etc.,…

When the test is requested with a certification, the certification center requests the software test to the test department with 'software test request form‘, ’product descriptor', 'user's manual' and software and is informed the written agreement by the test department. And the certification center make out the certification plan about the day's program, the process and the method from a test of the requested software to the final certificate.

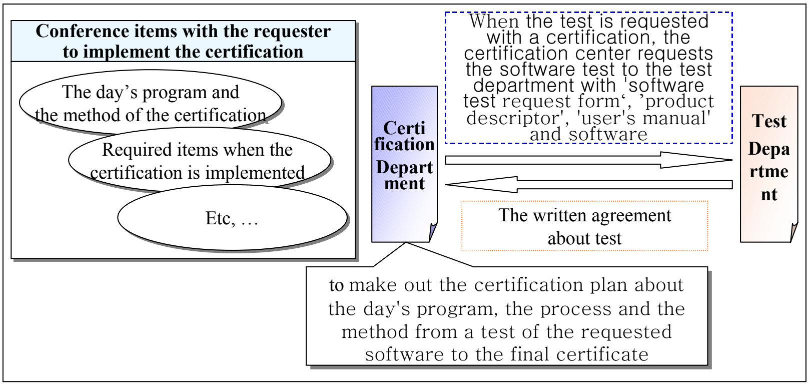

Conference items with the requester to implement the certification
The day’s program and the method of the certification
Required items when the certification is implemented
Etc, …
Certi fication Depart ment
When the test is requested with a certification, the certification center requests the software test to the test department with 'software test request form‘, ’product descriptor', 'user's manual' and software
Test Depa rtme nt
The written agreement about test
to make out the certification plan about the day's program, the process and the method from a test of the requested software to the final certificate

Fig. 3. Test Request and Establishment of the Certification plan

3.2.3 The Acceptance of the Test Results and Framing of Certification Assessment Data

When the test is ended, the certification team is delivered the test results from the test department. The test results must include the test grade to inform to the test requester. The certification team starts assessment task analyzing the test results and related data, and make out the certification/assessment data dependant on the criteria of certificate authority.

3.2.4 Certification/Deliberation Committee and Assessment Report

When the certification/assessment data were completed, the certificate authority holds a certification/deliberation committee and present agenda. The Certificate authority make out the assessment report based on the result of certification/deliberation committee and certification/assessment data and inform it to the applicant with the test grade after assessment.

When necessary, the certificate authority specifies inadequate items, a additional assessment, a test coverage and approval measures. If requester proof to take steps for improvement meeting all requirements in a given period, they redo the required assessment.

3.2.5 Certificate Delivery and Notice

After certification/assessment, if the software applied for certification is coincided with the certification condition and certification/deliberation committee decide resolve the certification, the certificate authority deliver a certificate and announce the certification.

The announced contents are as follows.

- The Category of Certification
- A firm name of those who is certified
- Software name and version
- Certificate number
- Certification year, month, day

3.2.6 Dissatisfaction and Administrative Appeal

When the client's dissatisfaction or demur is appealed, the certificate authority receives by a written application. If the received contents is slight, the certificate authority take immediate action by document. If they are important, the certification/deliberation committee treats it.

3.2.7 Follow Up Control

The certificate authority regularly control product sale, advertisement, production. If software affect to conformance of the certified software, the supplier must inform the certificate authority.

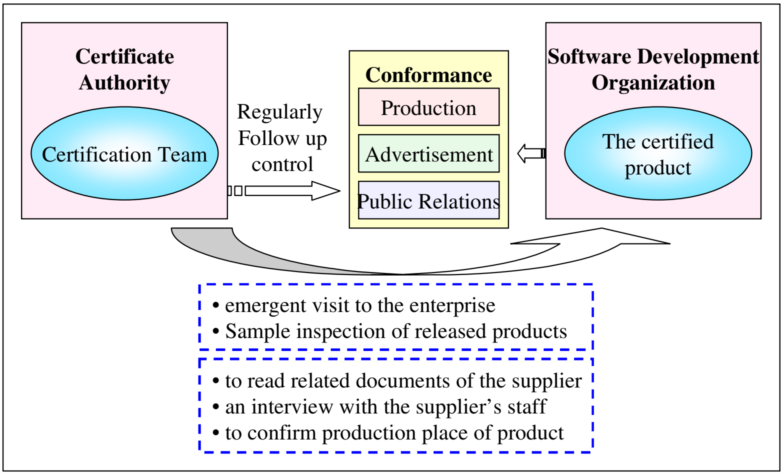

Certificate Authority
Certification Team
Regularly Follow up control
Conformance
Production
Advertisement
Public Relations
Software Development Organization
The certified product
• emergent visit to the enterprise
• Sample inspection of released products
• to read related documents of the supplier
• an interview with the supplier's staff
• to confirm production place of product

Fig. 4. Follow up control

4 Quality Model

In order to apply ISO/IEC 12119 to package software test, the quality model, which each item consisting of package software is to be applied to, should be organized.

4.1 Quality Model on Product Manual

Quality model on product manual among the constitutional elements of package software includes the items such as function, reliability, application, effectiveness, maintenance and graft, and those can be summarized as shown in Table 1.

Table 1. Quality Model about Product Manual

Quality Model	Concept
Functionality	Summary of functions, region value, security information
Reliability	Information for data storing process
Usability	User interface form, knowledge for product usage, identification of usage condition
Efficiency	Response time, processing rate
Maintainability	Explanation about maintainability
Portability	Explanation about Portability

4.2 Quality Model on User Document

Quality model on user document among the constitutional elements of package software includes the items such as perfection, exactness, consistency, understanding and easy summary, and those can be summarized as shown in Table 2.

Table 2. Quality Model about User Document

Quality model	Concept
Completeness	Product usage information, region value, installation-maintenance manual
Correctness	Correctness of document information, clearness of expression
Consistency	Integrity between documents, terms consistency
Understandability	User group have to understand
Easy summary	Easy summary about user documenet

4.3 Quality Model on Program and Data

Quality model on program and data among the constitutional elements of package softwares includes the items such as function, reliability, application, effectiveness, maintenance and graft, and those can be summarized as shown in Table 3.

Table 3. Quality Model about Program and Data

Quality model	Concept
Functionality	- Can Install according to the manual - similiar to all explanation in other document - not conflict with other documents - must be executed as specification
Reliability	- always controllable - data is not destructed
Usability	- understandability about all information of program - adequacy of error message information
Efficiency	- the explanation about efficiency is suitable
Maintainability	- the explanation about maintainability is suitable
Portability	- the explanation about portability is suitable

5 Development of Package Software Evaluation Metrics

Evaluation Metric for package softwares has the basis of ISO/IEC12119, and this study abstracted the Metric items that are applicable to package softwares from ISO/IEC 9126-2, 3, and modified and supplemented them. The details of developed Metric items are as shown in Table 4.

Table 4. Quality Model about Program and Data

Type of Me trics	The number of metrics	Remark
General requirements	10	Metrics about Identification and order
Product manual	20	Metrics about functionality, reliability, usability, efficiency, maintainability, portability
User document	12	Metrics about completeness, correctness, consistency, understandability, easy summary
Program & data	61	Metrics about functionality, reliability, usability, efficiency, maintainability, portability

5.1 Metric Index Table

This study built the Metric Index Table by product element consisting of package software, as shown in Table 5. The Metric Index Table on general requirements for package software is shown in the Table.

Table 5. Quality Model about Program and Data

Characteristics	Metric index	Type	Reference
Identification and order	1.1 identification of product manual	Y/N	ISO/IEC 12119
	1.2 identification of product	Y/N	ISO/IEC 12119
	1.3 Specification of supplier	Y/N	ISO/IEC 12119
	1.4 Specification of work	Y/N	ISO/IEC 12119
	1.5 Document for adequacy requirements	Y/N	ISO/IEC 12119

5.2 Construction of Metric Table

An example of Metric that is developed for the purpose of testing package softwares by product element is as shown in Table 6.

The example of Metric on general requirements for package softwares is shown in the Table.

Metric Table was developed, based on ISO/IEC 12119, and it was modified as suitable one for package softwares test by introducing some relevant items from ISO/IEC 9126-2, 3.

5.3 Decision of the Evaluation Marks Level and Judgment Standard on Metric Value

If the result value intends to have the meaning, it needs to decide the evaluation marks level on Metric value.

First, we define the evaluation marks level by deciding the number of range that Metric value has. The following example shows the case that defines 4 evaluation marks levels.

- A : excellent : satisfy all requirements
- B : good : satisfy almost requirements
- C : fair : not satisfy a part of requirements
- D : poor : not satisfy requirements

Table 6. A Sample of Metrics Table about General Requirements

Quality characteristics	Identification & order			
Specification of metric	**Detail item**		**Measu rement value**	**result**
1.1 identification of product manual computation : A value range : 0, 1	A	Is a unique document ID in product manual?		
		Example : name for product manual (function manual, product information, product pamplet, etc.		
problems				
1.2 identification of product computation : A value range : 0, 1	A	Is a unique ID in software product?		
		(Example) name, version, date, variant (Example) Variant : Enterprise version, Professional version, etc.		
Problems				

We can decide the range corresponding to evaluation marks level on each Metric value as follows.

- Measurement value 0<=X<=1
- X<0.6 : rating level D
- 0.6<=X<0.7 : rating level C
- 0.7<=X<0.8 : rating level B
- 0.8<=X : rating level A

Since the range of Metric measurement value is not always fixed, we decide it by considering the range of measurement value on each Metric. In this way, we can score evaluation marks according to evaluation marks level on each Metric, and if it acquires a certain level of evaluation marks, we get the final result by deciding the criterion to pass or fail. For example, supposing that they decide to purchase if the number of Metric, of which the evaluation marks level is B or above, is 95% or more, and if the test is applied to the several software as objects, we can decide to purchase the software that acquired the best result.

6 Conclusion

This study built quality test process for package software and developed Metric for testing package software by attempting to graft product evaluation process for evaluator in ISO/IEC 14598-5 into the standard of quality test for package software in ISO/IEC 12119, considering the features of package software.

If we firmly build evaluation system for package software with basis of the process for evaluator in ISO/IEC 14598-5, it is considered that we can build the effective evaluation basis for package software types that are made by many development organizations.

Regarding the study works after this, it needs to specify measurement methods on measured items of test Metric for package software, and push to develop effective quality test through tools.

References

1. ISO/IEC 9126, "Information Technology-Software Quality Characteristics and metrics-Part 1, 2, 3.
2. ISO/IEC 14598, "Information Technology - Software product evaluation - Part 1, 2, 3, 4, 5, 6.
3. Moller, K. H. and Paulish, D. J., "Software Metrics", Chapmen & Hall(IEEE Press), 1993.
4. Wallmuller, E., "Software Quality Assurance A practical approach", Prentice Hall, 1994.
5. 水野幸男, "ソフトウエアの綜合的品質管理", 日科技連出版, 1993.
6. 吉澤. 東. 片山, "ソフトウェアの 品質管理と生産技術", 日本規格協會, 1990. 5.
7. Ha-Sool Yang, Ha-Yong Lee, "Design and Implement of Quality Evaluation Toolkit in Design Phase”, KISS Paper(C), Vol. 3, No. 3, 1997. 6.
8. Hae-Sool Yang, “Quality Assurance and Evaluation of Hanjin Shipping New Information System”, Hanjin Shipping co., 1998. 9. 7.
9. Ha-Sool Yang, “Development of Sfotware Product Evaluation Supporting Tool”, ETRI Computer Software Technology Institute, 1999. 12.
10. Hae-Sool Yang, “Study on Quality Test and Measurement Criteria”, ETRI, Final Report, 2005. 11.
11. Hae-Sool Yang, “Construction of Quality Test and Certification System for Medical software”, Korea Food & Drug Administration, Final Report, 2005. 11.

Design of an On-Line Intrusion Forecast System with a Weather Forecasting Model

YoonJung Chung[1], InJung Kim[1], Chulsoo Lee[2], Eul Gyu Im[3], and Dongho Won[4,*]

[1] Electronics and Telecommunications and Research Institute
{yjjung, cipher}@etri.re.kr
[2] College of software, kyungwon University
cs1100@kyungwon.ac.kr
[3] College of Information and Communications, Hanyang University
imeg@hanyang.ac.kr
[4] Information Security Group, School of Information and Communication Engineering,
Sungkyunkwan University
dhwon@security.re.kr

Abstract. Information protection for information systems is the major concern for most of the institutes, but there are a limited number of activities for the prevention of intrusion. Though each institute establishes and operates information protection solutions such as information security control systems, counter-measures against intrusions are generally applied only after intrusions have taken place in most cases. Delayed counter-measures lead to delays in damage recovery as well as failure of timely actions to mitigate the damages. In this paper, we propose the design of an online intrusion forecast system using a weather forecasting model, allowing administrators to minimize the effects of damages in advance through an online intrusion prediction of the probable vulnerability and risks. Both the information from the sensors of information security control systems and the profiles of the information system assets are used to analyze vulnerabilities and to predict intrusion routes and the scope of damages.

Keywords: Intrusion, Weather Forecasting, Forecast, Damage Propagation, Information Security control system (ISMS).

1 Introduction

The growth of the Internet and the development of information technologies have accelerated both the complexity and the vulnerability of information systems. In particular, there have been ever-increasing threats from malignant codes that are capable of damage propagation on a large scale by exploiting the security deficiencies of information systems and the network infrastructure. Therefore, to protect system assets effectively from malicious attacks, a security administrator should be able to identify and predict the potential vulnerability and threats faced by information systems. Though information on the vulnerabilities of information systems is released on a

[*] Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 777 – 786, 2006.

daily basis [1] [4], the direct effects of the vulnerabilities on the information systems operated at institutes are not fully understood. The major reason for this is the virtual impossibility for an administrator of analyzing and taking action against every new risk. Therefore, for selected vulnerabilities, an administrator shall analyze possible infection routes and damage propagation using the profiles of the information assets and their probable vulnerabilities without any definite procedures or mechanisms [2] [3]. Recently, some institutes have started to operate information security control systems for real-time prevention, analysis, and counter-measures so that cyber attack damages can be minimized. The information security control system recognizes status of servers, network systems, application jobs, and the user's input and output, analyzes information flow within the organization, and takes appropriate actions for any risks. As shown in Fig. 1, the information security control system keeps track of networks and servers within the organization, and their traffic patterns. Once the alarm has been triggered the system executes rapid and appropriate counter-measures using the analysis results of abnormal conditions. These tasks allow the information security control system to gather various kinds of information about the servers and the networks, and as a result, the system is able to identify types of vulnerability and any threats. Then, the information security control system can inter-operate with other security elements such as firewalls, VPN, IDS, IPS and so on.

For these tasks, an agent is installed and operated in the information security control system. We propose an architectural design to make use of these agents to collect information from the profiles of the assets, to transmit the collected data to the prediction system, and then to suggest a design for the intrusion prediction systems based on the collected information.

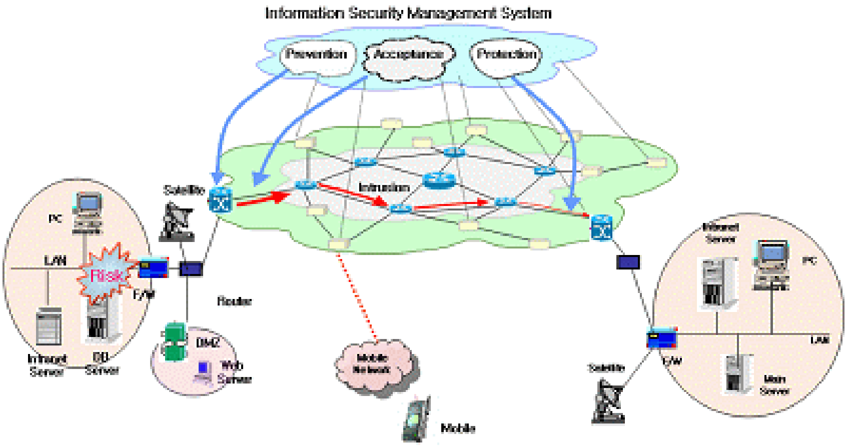


Fig. 1. Structure of the Information Security Control System

Potential vulnerabilities and threats to the information systems are classified into various kinds of intrusions, and a prediction algorithm is developed for intrusions that have typical propagation features and cause significant damages. A weather prediction model is used to provide an online intrusion forecasting. This allows an administrator to predict possible intrusion routes and their scopes of damages as soon as the alarm is triggered. Using the proposed prediction algorithm, quick and timely counter-measures are possible to reduce the damages.

2 Related Work

Studies on the prevention of intrusions have so far been conducted mainly for risk analysis [6] [7] and damage propagation [8] [9]. These approaches are based on a procedure consisting of several phases, such as analysis of assets and threats, calculation of risk levels, analysis of a proposed protection plans, evaluation of remainder risks, and establishment of final protection plans. This procedure is, however, extensive and complex analytical processes, making it very hard to implement and satisfy requirements of real-time managements for information systems. To resolve these problems, studies have been performed to develop a mechanism that can automatically identify changes of risks that are caused by changes of information assets [10]. But, in that mechanism, an administrator requires to provide the system with a detailed forecast of vulnerability and threats.

In this paper, we propose a model that predicts the intrusion routes to information systems using the weather prediction model [5] and analyzes the affects of intrusions. The proposed model is able to estimate potential damages from the predicted intrusions, and to identify counter-measures to minimize the damages. It should be noted that there are differences between weather prediction and intrusion prediction:

- External factors such as rain, snow, and hurricanes may proceed in temporal and spatial sequence, while intrusion takes place in the information system in an omni-directional sense.
- Changes of weather can be sensed in advance, while occurrences of intrusions cannot be sensed or identified in advance. Identification of an intrusion is only possible after the analysis of the dimensions of the risks posed by a system's vulnerability and threats.
- Weather prediction is done for an extensive period of time and protection measures can be established in advance, while intrusions occurs simultaneously after a usual short dormant period, and counter-measures according to prediction may not be feasible.
- Weather incidents have a very limited number of types, while every new intrusion occurs in a new form or pattern.

However, intrusion prediction is similar to weather prediction from the aspect that an intrusion prediction mechanism generates possible intrusion routes and damage areas by analyzing information from many sensors in the networks. The most widely used weather prediction models are as follows:

- Numeric model;
- Statistical models such as the persistency model, Markov model, and autoregressive process model;
- Response model through the convolution of the input and output systems;
- Artificial neural network model predicting weather as the output for new input according to the features of the system through existing case studies.

Table 1. Variables and constants used by DLM

Symbol	Definition
Y_t	Observed value at time t
F_t	Output vector at time t (forecast factor)
θ_t	Dynamic coefficient vector at time t (status vector, weight)
G_t	Transition matrix describing change of θ_t at time t
m_t	Mean value of θ_t at time t
C_t	Distribution of θ_t at time t
v_t	Output error
V_t	Distribution of output error v_t
w_t	Internal error
W_t	Distribution of internal error w_t
D_t	Information obtained from both the observed values up to t time and values calculated from the models
Φ	V^{-1}

Among the above-mentioned models, the numeric model (with a wide range of prediction) is one of the most popular models. However, since certain features of the model do not allow the full reflection of complex topographical characteristics and it requires too much time to calculate numeric values for the model, the usage of the model is limited. Therefore, statistical methods are widely used for the real-time weather prediction. Though incapable of illustrating physical properties and dynamics of weather, these methods are capable of calculating prediction information with fewer errors on non-linearity when weather moves to other areas.

We employ the statistical model to design an intrusion forecasting system. This model illustrates changes in status as functions of time. The dynamic linear model (DLM) consists of the following equations, and each function or variable is defined as in Table 1.

$$Y_t = F_t{}'\theta_t + v_t, \qquad (\textit{Output equation}) \tag{1}$$

$$\theta_t = G_t\theta_{t-1} + w_t, \qquad (\textit{Status equation}) \tag{2}$$

$$v_t \sim N(0, V_t),\ w_t \sim T_{n_{t-1}}(0, W_t)$$

$$\theta_t \big| D_t \sim T_{n_t}(m_t, C_t),\ \Phi \big| D_t \sim G\left(\frac{n_t}{2}, \frac{n_t s_t}{2}\right)$$

DLM generates dynamic coefficient vectors with the initial values given in each time slot and status equations, and generates output values with output equations.

Up-to-date states are reflected in prediction results, and up-to-date states are calculated through a dynamic circulation process that estimates optimal dynamic coefficients at time t+1 using various factors such as observed values at time t, types of transition matrices, and error distributions.

3 Intrusion Prediction System

Our proposed intrusion prediction system for information systems employs the agents of the information security control system. The current security control system is connected to Firewalls and IDS for real-time management; however, the security control system is unable to predict or analyze potential intrusion routes arising from virtual threats and/or system vulnerability. Thus, this system is not suitable for intrusion forecasting. However, prediction factors can be obtained using the agents used in the security control system. The following kinds of agents are deployed in networks for our intrusion prediction system:

- Core agent: managing the agents installed inside the assets, and predicting the status of the assets.
- Sensor agent: installed in each asset to check changes and status of the assets.
- Transmitter agent: transmitting information from the sensor to the control system.
- Profile reader agent: storing and managing asset information from the sensor.
- Comparator agent: comparing information from the sensor with existing information in order to identify abnormal events or emergency situations.
- Predictor agent: providing information on concerned assets upon occurrence of intrusions due to virtual threats and system vulnerability.

The difference between these agents and those of the existing control system is that the agents are installed on the information protection systems as well as other systems like terminals, servers, and network systems in order to collect information about any changes in assets and systems. Fig. 2 illustrates the example configuration of the agents.

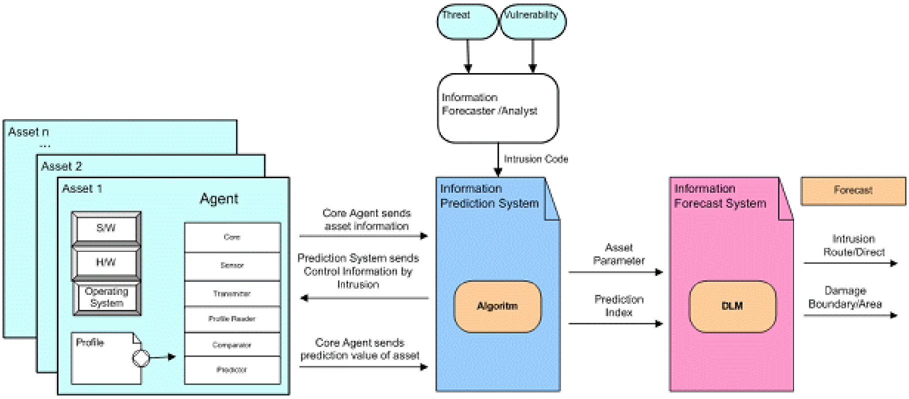


Fig. 2. Configuration of prediction agents

To have better prediction, the agents have to obtain detailed information from the information systems. The agents operate in either of the following modes: a normal prediction mode during normal operations, and a special prediction mode for online intrusions. If the normal prediction mode is set, prediction tasks are performed in accordance with predefined schedules, with few changes in the information systems during operations when abnormal events are identified in the information systems over a certain period. The special prediction mode is set when prediction tasks are outside the scope of the normal prediction mode. For instance, if activities of a worm or a virus are suspected or hacking on a system is suspected, the special prediction mode is set, and the system tries to collect real-time information concerning intrusion accidents by focusing

Table 2. Information profile

Factor	Features of analysis data	Calculation technique/cycle
Threat (t_a)	Threat to system	Upon occurrence of threat
Vulnerability (v_a)	Vulnerability of systems	Upon occurrence of vulnerability
Threat vector (r_a)	Existing risk information	Resultant value of previous risk
Information asset (a)	Location and capability of assets	System purchase
Network asset (n)	Networks configuring a system	System installation
Task type (p)	Task types used by a system	System upgrade
Inter-operation (I)	Inter-operation between assets	System inter-operation
Utilization quantity (U)	Utilization quantity and allowed quantity of assets	Daily
Period of use (C)	Duration of assets' use, and user's access	Upon acquiring user's authorization
Prevention/detection (P_1)	Level of intrusion detection	Upon introduction of information protection system
Analysis (P_2)	Level of intrusion analysis and type classification	
Recovery (P_3)	Maximum allowed duration of recovery upon intrusion	
Confidentiality (S_1)	Confidentiality of information system	Upon establishment of information protection strategy
Integrity (S_2)	Integrity of information system	
Availability (S_3)	Availability of information system	
Backup/disaster (b)	Redundancy and backup status against intrusion	During backup
Management (M)	Analysis of policies, guidelines, organization, human resources, and training	Performance of risk analysis

on data from suspected areas of the information systems and obtaining temporal and spatial data. The objectives of the normal and the special prediction modes differ from each other, and types and a calculation period also vary depending on calculation natures of the information systems. Information from the agents contains raw data (level 1.0 data) of assets and a network structure, level 1.5 data such as job types, interoperation, utilization quantity and period of use, and level 2.0 data regarding operations and managements of the information protection systems.

The normal prediction mode calculates 17 types of basic analysis data as shown in table 2, and the special prediction mode provides information on network traffic, service rates, and access counts upon intrusion. Rapidity is a prerequisite for data generated in the special prediction mode. Information assets include hardware, software, and data. These assets are significantly affected by vulnerability and threats. Hardware includes terminals, servers, and databases; and software includes commercial software and developed software; data includes file data and database data. These primary categories are further classified into the secondary and the detail categories, and information in each category is utilized to map vulnerability and threats. For example, the network assets (n) in Table 2 include network equipments, network maps, and the information protection systems. The network equipments are further classified into routers, hubs, and switches; the network maps into WAN, LAN, and DMZ; and the information protection systems into Firewalls, IDS, and secure OS. Information about potential routes of intrusion can be obtained from mapping between threats.

Based on data from the agents with consideration of the relationship between the agents, the algorithm of the information prediction system is configured as shown in Fig. 3. The input data of the algorithms is classified into two kinds of data: the static input data that does not have significant changes over a certain period, and the dynamic data that is frequently changed. The static input data includes models of servers, computers, and routers, and versions of operating systems. Since asset information that is subject to change after a certain period significantly affects the precision of results, this kind of asset information must be frequently checked and reflected in the analysis process. The dynamic input data is acquired in real-time, and examples are level 2.0 data, special announcement data, and so on.

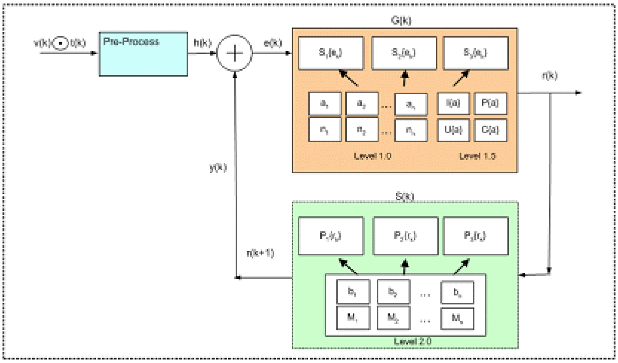


Fig. 3. Algorithm of the information prediction system

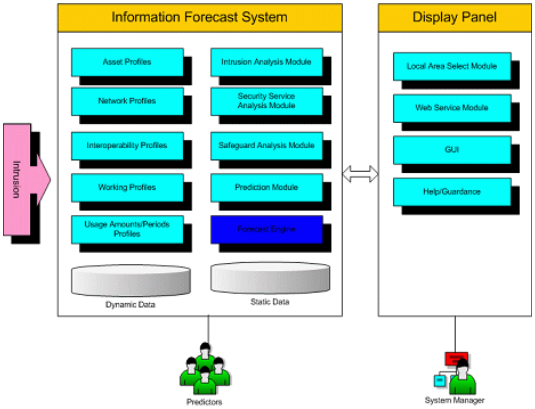


Fig. 4. Configuration of the forecasting system

As illustrated in Fig. 4, pre-processing of the input data is essential prior to intrusion analysis and the generation of prediction data. Pre-processing of the input data is performed to find threats and vulnerability in the input data in order to define the level and category of intrusion, since damage to the information systems is not caused by threats and vulnerability but by the occurrence of intrusions. Therefore, threats and vulnerability are combined and analyzed to find potential intrusions, and relevant exploit codes are also identified and analyzed in advance. The feasibility of identifying threats and vulnerability is provided as probability values, and the results are used as input values for intrusion detection.

Once the profile of the information system has been defined, inter-operation between the task types and the assets is used to calculate flows of intrusions. Further, utilization statistics about information assets is also used to calculate the speed of intrusion propagation. A risk vector is used to analyze changes of the risks in the course of a certain time span, and the risk level of the information system is determined. The degree of the information system safety is affected by various factors, such as how well the information protection system is operated from security's point of view. The propagation of intrusions can be limited and propagation routes can be simplified through processes like prevention, detection, and recovery. Aspects of intrusion analysis are classified into confidentiality, integrity, and availability, and severity of an intrusion is determined by comparing the analysis results with the information protection level required by the information system. All of the analysis data is, in principle, calculated in the normal prediction mode cycle, and data generated from the special prediction mode is adjusted to the analysis data.

4 Intrusion Forecasting System

The intrusion forecasting system is designed based on the prediction results, by making use of the DLM described in Section 2. Detailed configuration by phase are as follows:

[Phase 1] Setting initial values: initial values are set to determine weights of forecasting factors up to the initial time.

[Phase 2] Calculating weights for forecasting factors:

- ▶ The reduction factors are defined to reduce errors in prediction values, and the measured values with increments of 0.01 between 0.1 and 1 are increased and applied to calculate prediction values. This upgrading process is repeated, and results are compared with the measured values.
- ▶ The minimum value of the results calculated using the prediction values and the measured values in the upgrading process is selected as the optimal reduction factor.

[Phase 3] The prediction value is calculated by applying weights from [Phase 2] to the measured and the forecast data at the analysis time.

The intrusion forecasting system processes the prediction values provided in default, and generates forecasting data through the decision-making model, and store the results in a database. The produced data is provided to administrators responsible for the information systems. Fig. 4 illustrates the configuration diagram.

The forecasting factors, the most significant factors in the forecasting system, are produced as follows. Forecasting intrusions are defined for two cases: the situation of actual intrusions and that of probable intrusions. In case of the probable situation of intrusions, sufficient time is allowed for responses. However, In case of actual intrusions, prevention opportunities are limited. Therefore, the probability of forecast should be calculated considering the intensity of the threats and their predicted occurrence frequency. We determine the actual occurrences of an intrusion when the intrusion exceeds a threshold, and define the intrusion probabilities.

5 Conclusion

We suggest a design for an online intrusion forecasting system. We also suggest a model for predicting and forecasting any intrusions in dynamic manners in the course of a specific time span by analyzing the potential propagation of vulnerability and threats. Comprehensive prediction and analysis of damage propagation are pursued based on the model for the information systems. We utilize the suggested models and systems to validate their feasibility for effective forecasting through case studies. In this study, we provide security administrators with measures for predicting security damage accidents, establishing early-warnings and security controls against any threats, and minimizing damages to organizations and institutes.

Special attention shall, however, be paid to encrypting and storing the profile information of information assets to prevent the disclosure of any information to potential hackers. In addition, we should also consider further studies for security protocol and mutual certification techniques upon exchange of information among the agents and the prediction systems.

References

[1] CERT CC, http://www.cert.org

[2] Cliff C, Zou, Weibo Gang, Don Towsley, "Code Red Worm Propagation Modeling and Analysis", 9th ACM Conference on Computer and Communication Security (CCS'02), Nov.18-22, Washington DC, USA, 2002.

[3] Thomas Dubendorfer, Arno Wagner, Bernhard Plattner, "An Economic Damage Model for Large Scale Internet Attacks," Proceedings of the 13th IEEE International Workshops on Enabling Technologies Infrastructure for Collaborative Enterprise (WET ICE'04) 1524-4547/04.

[4] Open Web Application Security Project, http://www.owasp.org

[5] Pikoulas, J.; Buchanan, W.J.; Mannion, M.; Triantafyllopoulos, K. "An agent-based Bayesian forecasting model for enhanced network security," Engineering of Computer Based Systems, 2001. ECBS 2001. Proceedings. Eighth Annual IEEE International Conference and Workshop, April 2001.

[6] Hoh Peter In, Young-Gab Kim, Taek Lee, Chang-Joo Moon, Yoonjung Jung, Injung Kim, "Security Risk Analysis Model for Information Systems," LNCS 3398, Systems Modeling and Simulation: Theory and Applications: Third Asian Simulation Conference, AsianSim 2004.

[7] Injung Kim, YoonJung Jung, JoongGil Park, Dongho Won, "A Study on Security Risk Modeling over Information and Communication Infrastructure," SAM04, pp. 249-253, 2004.

[8] Young-Hwan Bang, YoonJung Jung, Injung Kim, Namhoon Lee, GangSoo Lee, "Design and Development of a Risk Analysis Automatic Tool," ICCSA2004, LNCS 3043, pp.491-499, 2004.

[9] Yoon Jung Chung, InJung Kim, NamHoon Lee, Taek Lee, Hoh Peter In, “Security Risk Vector for Quantitative Asset Assessment”, Volume 3481 / 2005, Computational Science and Its Applications – ICCSA 2005: International Conference, May 9-12, 2005.

[10] InJung Kim, YoonJung Chung, YoungGyo Lee, Dongho Won, "A Time-Variant Risk Analysis and Damage Estimation for Large-Scale Network Systems," ICCSA2005, LNCS3043, May 2005.

[11] Forum of Incident Response and Security Teams, http://www.first.org

Goal Programming Approach to Compose the Web Service Quality of Service

Daerae Cho[1], Changmin Kim[2], MoonWon Choo[3],
Suk-Ho Kang[1], and Wookey Lee[2,*]

[1] Dept. Industrial Engineering, Seoul National University
{drcho, shkang}@ara.snu.ac.kr
[2] Div. Computer Science, Sungkyul University
[3] Div. Multimedia, Sungkyul University
{wook, kimcm, mchoo}@sungkyul.edu

Abstract. As business environments are changed and become complex, a more efficient and effective process management are needed. More and more enterprises and organizations are recently trying to build flexible and integrated information systems with web services in order to satisfy the changing needs of customers. The web Service can currently be recognized as a new alternative for integrating the scattered information assets within an enterprise or an organization. Due to the increasing number of Web Service applications and the service suppliers, however, the customers are confronted with the problem of selecting the most suitable Web Service. In this paper the new methodology for marshaling the composite Web Service satisfying Web Service QoS goals is suggested. This provides a theoretical basis from which a goal programming model is identified by which the web service QoS can be quantified.

1 Introduction

Composition of Web Services are currently received much interest to support business-to-business or enterprise application integration. Enterprises and organizations have been trying to build flexible and integrated information systems in order to satisfy rapidly changing needs of customers, in incorporating the conventional organization and information systems. EAI (Enterprise Application Integration), the one of such an effort to integrate the various solution packages, could not handle the steadily increasing demand for the cooperation mechanism reflecting the e-business environment [1]. Also it is not so easy to integrate the business inter-processes, since the emergence of value may be preconditioned by the intrinsic incorporation of processes shared by partners [2]. Since intrinsic difficulties are caused by heterogeneity, locations, scalability, modification, etc. exist in performing that task, Web Service can currently be recognized as a new alternative for integrating the scattered information assets within an enterprise or an organization.

[*] Corresponding author.+82-31-467-8174.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 787 – 797, 2006.

Web Services support the interaction of business partners and their processes by providing a stateless model of atomic synchronous or asynchronous message exchanges. They may be identified by URI, as a software system whose public interfaces and binding are defined as XML-based messages, and can be defined and supported in the Internet [8]. A set of appropriate Web Services, which is platform-independent software component and are available in the distributed environment of the Internet, can be assembled into applications. Seamless composition of Web Services has enormous potential in streamlining business-to-business or enterprise application integration. Above all, Web Service paradigm is strongly supported by the influential de-facto standard organizations like IBM, OASIS, ORACLE, SUN, Microsoft IOS as the promising developing framework of software architecture [7][9]. However, due to the increasing number of Web Service applications and the service suppliers, the customers as well as business communities are confronted with the problem of selecting the most suitable Web Services.

Complex Web Service composition can be done after finding out the multiple service suppliers in UDDI that can perform the tasks in service processes. In this paper, non-functional factor, QoS is considered in setting the criteria for finding out the optimal suppliers, who can provide the same functions. The setting the criteria for QoS can be typical MCDM(Multiple Criteria Decision Making) problem which is NP-hard as a special case of Knapsack Problem [12] and the determination of selection will be the determining variable, 1-0 Integer problem is recognized as the best choice to solve the composition here.

The rest of paper is organized ad follows. Section 2 is for related works. Then QoS for Web Services and QoS modeling are discussed. We conclude with experimentation and future work.

2 Related Works

WFMS (Workflow Management System) provides the theoretical foundation for reengineering the enterprise structure and automating the business efficiently [3]. Workflow is considered as the standard computing model for interoperating processes and exchanging the information in the Web-based environment [5]. When composing Web Services, workflow back-ended approach suggests the ordering method for considering QoS. First, the criteria for process quality evaluation are determined. Then the qualities such as execution duration, cost, reliability, etc. are estimated. Secondly, the qualities of Web Service process are evaluated based on the quality of selected tasks using the structural information of process. That is, the quality of a process is estimated by the repeated reduction of the serial and parallel blocks embedded in a process into one task. To apply this method, the quality dimension of the task in a process should be estimated. J. Cardoso suggests the approach to estimate the task quality, applying it to Fault-Tolerance System, Network System, etc. [3]. Workflow back-ended approach can calculate the quality according to process structure, being adaptive to the structural differences. However, it is not easy to apply this method when the number of service suppliers is increasing and to find the optimal suppliers because of the limitation of simulation approach taken to search the optimal services.

Process-based Web Service composition is considered as the effective method for integrating the heterogeneous and distributed applications [1]. In UDDI registry the enormous service suppliers are resided and their status of registration can be changing in real-time. Therefore, it is still challenging to select the service supplier considering the QoS during complex Web Services are executing [6]. Basically, selecting Web Service suppliers could be done by considering process structure showing AND structure and XOR structure. Zeng assumes that XOR branching can be defined as the possible path after separating this branching and the selection decision of path is determined by the task scheduled before the XOR branching [2]. He defined the term 'execution plan' as a set of service suppliers being able to perform the tasks existing within the 'execution path'. In this perspective, the problem to find the optimal service suppliers for Web Service composition can be transformed into the problem to find out the optimal execution path. The QoS for each execution plan is represented as the linear equation, which is claimed to be used to find out the optimal execution path after scaling using Linear Programming. It is a Zeng's strong points that his method guarantees the global optimization and could be used as a general approach in term of QoS evaluation criteria under the condition that it is linearly formulated. However, his assumption about the branching condition of XOR structure execution into execution paths can not be applied to the other types of process structure. Also the criteria formulation with arbitrary weights using linearly programming may not be so practical.

3 QoS for Web Services

Process structure is said to be the ordered relations defined between the unit tasks consisted of processes. In this paper, SWR(Stochastic Workflow Reduction) algorithm, which is the approach to reduce the predefined process structures into single task to estimate the process quality, is adopted. Workflow process structures are classified into several types using the concept of 'block', which is further classified as serial and parallel block. Serial block has one path along which no branching and combining is not happened. Parallel block has multiple paths between the branching unit task (a_S) and combining unit tasks (a_M).

3.1 QoS Requirements for Web Services

The requirements of Web Service QoS proposed by IBM include the non-functional attributes like the process time of Web Services, cost, reliability, etc.. In this paper, the criteria for selecting the Web Service partners is set based on the QoS of services requested by consumers, which can be evaluated quantitatively as follows;

- Execution Duration – is the time elapsed from the customer request of service to the receipt of response from the Web Service supplier. Hence, it may be composed of the request time, service time and the time needed for sending the results.
- Execution Cost – is defined as the cost to be paid for the execution of Web Services.

- Reliability – is the probability of receiving the processing result within the expected duration time set randomly, when the Web Service is requested. It may be considered as the measure to guarantee the message transmission between customer and service supplier.
- Availability – is the criteria for evaluating an immediate availability of a Web Service. It can be computed as the ratio of the service time to the total time of observation.

$$\text{Availability}=\frac{\text{<Up Time>}}{\text{<Total Time>}}=\frac{\text{<Up Time>}}{\text{(<Up Time>+<Down Time>)}} \cdots\cdots\cdots\cdots\cdots (1)$$

- Reputation – is the factor for evaluating the service reliability based on the customer's experience. In this paper, it is defined as the average of the final customer's evaluation on the Web Services.

$$\text{Reputation}=\frac{1}{n}\sum_{i}^{n}\text{<user's Rating>} \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (2)$$

3.2 Hypothesis

The plausible selection of Web Service suppliers is set up as the determining variable. From this perspective, AND Structure and XOR Structure are taken into consideration in the case of parallel structure of Web Service process. The evaluation criteria for QoS can be formulated according to the each process structure, then the results are combined. Each QoS criteria can be an objective function, so there come out multiple objective functions, which are the constraints of Goal Programming to minimize the deviation from the QoS demanded by customer. The formulation of criteria is done under the following assumptions.

- Independency: all tasks resided in process are mutually independent.
- Trustfulness: the quality level of services is reliable
- Active Selection: Web Service customer can arbitrarily select a path among the paths characterized by XOR branching.

4 WS QoS Modeling

4.1 Notation

The problem defined in this paper is to find the optimal Web Service suppliers to perform the tasks in process, when composing the complex Web Services. Hence, the determining variable can be characterized by the plausibility of selection of particular Web Service suppliers.

x_{ij} : the selection value j^{th} service supplier among i^{th} task (0: unselected, 1: selected),

S_i : the set of all service suppliers in i^{th} task,

S^{xor_n} : n^{th} XOR set,

$S_i^{xor_n}$: the set of all service suppliers in i^{th} task within n^{th} XOR set

The qualities characterized by service suppliers performing particular task is represented as follows;

r_i : the reliability of ith task

c_{ij} : the cost of ith task performed by jth service supplier

t_{ij} : the execution duration of ith task performed by jth service supplier

r_{ij} : the reliability of jth service supplier in the ith task

av_{ij} : the availability of jth service supplier in ith task

re_{ij} :the reputation of jth service supplier in the ith task

Reliability, Availability, and Reputation are non-linearly expressed and formulated by regarding the quality of service suppliers to the quality of selected task. So the quality of a task is represented as follows:

r_i : the reliability of i^{th} task

av_i : the availability of i^{th} task

re_i : the reputation of i^{th} task

T_i : the execution duration of process to i^{th} task

T_{start} : the initial time of process

T_{end} : the ending time of process

The level of QoS demanded by customer is represented as follows:

C : the execution cost of complex Web Service requested by customer
T : the execution time of complex Web Service requested by customer
R : the reliability of complex Web Service requested by customer
Av : the availability of complex Web Service requested by customer
Re : the reputation of complex Web Service requested by customer

Based on the structural information mentioned above, the Web Service is defined as below.

Definition 1. All Web Services existent from the nth XOR set S^{xor_n} to kth path is Φ^n(k).

In case of XOR set S^{xor_n} , there exist k paths. The possibility of selection of each path is defined as w_k^n . That is, if w_k^n is set to 1, the kth path in nth XOR set is selected. Otherwise, it is set to 0 (not selected). Based on the definitions above, the additional constraints within XOR structure are as follows:

$$\sum_{j \in S_i^{xor_n}} x_{ij} = w_k \text{ , where } \forall x_{ij} = 0 \text{ or } 1 \text{ and } x_{ij} \in \phi(\text{k}) \text{ .... } (3)$$

$$\sum_{i}^{k} w_i = 1 \text{, where } \forall w_k = 0 \text{ or } 1 \text{.................................... } (4)$$

4.2 Nested XOR Structure

In the case of nested AND structure or XOR structure within XOR structure, the nested structure is performed depending on the resultant selection of nested paths. This can be theorized as follows:

Theorem 1. If the AND structures are nested within the kth path (Φ^n(k)) of XOR structure, the execution of tasks in the AND structure is performed depending on the resultant selection of nested paths ($\sum_{j \in S_i} x_{ij} = w_k^n$).

Theorem 2. If the n+1th XOR structure($S^{xor_{n+1}}$) is nested within the kth path of nth XOR structure, the execution of n+1th XOR structure is performed depending on the resultant selection of kth path of nth XOR structure ($\sum_{i} w_i^{n+1} = w_k^n$).

If another XOR structure is nested within XOR structure, the execution of n+1th XOR structure is performed depending on the value $w^n(k)$ of by Theorem 2. Hence, the additional constraint imposed on the n+1th XOR structure is as follows:

$$\sum_{i}^{3} w_i^{n+1} = w_k^n \text{ .... } (5)$$

4.3 Quality-Driven Web Service Selection

As mentioned above, Goal Programming is used for minimizing the QoS deviation. The deviation variable and its penalty are described as follows:

S_i^+ = Amount by which we numerically exceed the ith goal

S_i^- = Amount by which we numerically under the ith goal

P_l = The penalty for un-fulfillment of the ith goal

Optimal Web Service suppliers which are process-independent, are picked using the following equations under the consideration of QoS.

Min $S_1^+ + P_2 S_2^+ + P_3 S_3^- + P_4 S_4^- + P_5 S_5^-$

Subject to

$$\sum_i \sum_{j \in S_i} c_{ij} x_{ij} + S_1^- - S_1^+ = C \qquad (6)$$

$$T_{end} - T_{start} + S_2^- - S_2^+ = T \qquad (7)$$

$$\prod_i r_i + S_4^- - S_4^+ = R \qquad (8)$$

$$\prod_i av_i + S_3^- - S_3^+ = Av \qquad (9)$$

$$\frac{1}{n} \sum_i \sum_{j \in S_i} re_{ij} x_{ij} + S_5^- - S_5^+ = Re \qquad (10)$$

$$\sum_{j \in S_i} x_{ij} = 1,\ \forall x_{ij} = 0 \text{ or } 1 \qquad (11)$$

$$n = \sum_i \sum_j x_{ij} \qquad (12)$$

$$\sum_{j \in S_i} x_{ij} = \begin{cases} 1 \\ w_k^{n-1}, \text{ where } x_{ij} \in \phi^{n-1}(\mathrm{k}) \end{cases} \qquad (13)$$

$$\sum_i w_i^n = \begin{cases} 1 \\ w_i^{n-1}, \text{ where } S^{xor_{n-1}} \in S^{xor_n} \end{cases} \qquad (14)$$

$$\forall x_{ij} \text{ and } w_i^n = 0 \text{ or } 1 \qquad (15)$$

Equation (6) computes the execution cost by summing the total cost after selecting a service supplier from each task. The execution time in Sequential structure corresponds to the execution time of the task taken by the selected service supplier. This equation is modified as equation (7) by considering AND structure, computing the execution time elapsed along the Critical Path using PERT/CPM algorithm. Equation (8) and (9) computes the reliability and availability, multiplying reliability of particular service supplier performing task with availability. Equation (10) represents the reputation of Web Service, averaging the reputations of tasks. Equation (13) claims that only one service supplier should be selected for performing task and the result comes out depending on the resultant selection of the path that is the only path in XOR branching. Equation (14) claims that only one XOR structure should be selected and the result comes out depending on the resultant selection of the path which is nested in XOR structure.

5 Experimentation

The experimental scenario is devised as shown in Figure 1. The purchasing process is executed using Web Services, and each process has 11 tasks.

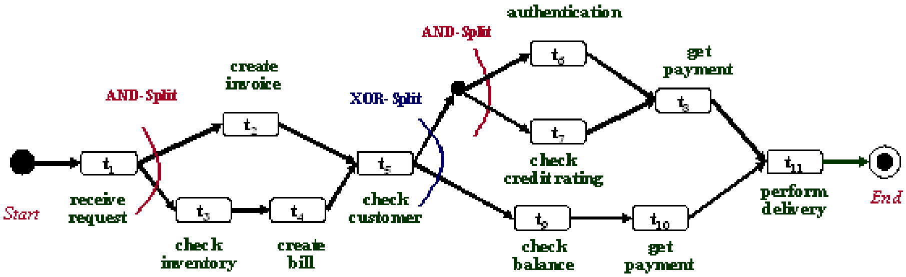


Fig. 1. Purchasing process for simulated scenario

After receiving the orders form customer (t_1), invoice is issued (t_2), inventory is checked and the bill is prepared (t_3). After all previous tasks are over, the customer is identified and the terms of payment are confirmed (t_5). If the payment is to be processed by credit card, the customer's identification is verified (t_6) and the credit status is checked (t_7), then the payment is approved (t_8). If the payment is to be processed by bank account, the balance is checked (t_9) and the payment is approved (t_{10}). The product is delivered to customer after the payment is confirmed (t_{11}). The customer, who are involved in this purchasing process, is assumed to make SLA (Service Level Agreement) shown in Table 1 with the service supplier of purchasing process. Two service suppliers exist in each task whose QoS are generated randomly. Table 2 shows the result derived by using LINGO 7.0 [7, 8].

Table 1. SLA for QoS

	SLO (Service Level Objective)	Penalty
Execution Duration	60 s	$5/s
Execution Cost	$ 800	$ over cost
Reliability	95%	$ 100
Availability	95 %	$ 50
Reputation	8	$ 50

The goal of this experimentation is to evaluate the plausibility of Goal Programming. Table 3 (b) shows the result after applying the LINDO7.0, saying that the execution cost and reputation do not meet the QoS demanded by a customer. If the service suppliers are chosen as Table 3(a), the execution cost will be exceeded by $0.6 and the reputation is lowered by 0.1 after composition.

Table 2. QoS for Web Service suppliers QoS

		duration	cost	reliability	availability	reputation
Task 1	X_{0101}	6.3	91.5	0.993	0.9985	10
	X_{0102}	9	75.4	0.9942	0.9948	6.1
Task 2	X_{0201}	19.8	156	0.9945	0.9967	7.4
	X_{0202}	17.5	195.2	0.9909	0.9973	6.1
Task 3	X_{0301}	8.4	91.2	0.9973	0.9985	7.1
	X_{0302}	8	94.4	0.9969	0.9924	9.1
Task 4	X_{0401}	9	80.5	0.995	0.9948	7.9
	X_{0402}	10.3	89.5	0.992	0.9987	6.5
Task 5	X_{0501}	7.6	91.6	0.995	0.9928	8.2
	X_{0502}	8.2	74.5	0.9976	0.9974	8.2
Task 6	X_{0601}	8.7	82.5	0.9961	0.9939	9.7
	X_{0602}	7.5	91.4	0.9947	0.9923	7.3
Task 7	X_{0701}	7.4	83.4	0.9932	0.9919	8.8
	X_{0702}	9.4	73.4	0.9999	0.9929	8.6
Task 8	X_{0801}	7	81.2	0.9948	0.9917	8.9
	X_{0802}	6.5	70.4	0.9917	0.9918	7.3
Task 9	X_{0901}	14.7	149.1	0.9984	0.9915	7.1
	X_{0902}	10.2	133.8	0.9971	0.9985	9.6
Task 10	X_{1001}	12.8	121.5	0.9904	0.9909	9.5
	X_{1002}	12.8	104.4	0.9912	0.9994	7.2
Task 11	X_{1101}	6.8	93.5	0.9994	0.9978	6.8
	X_{1102}	9.3	90.4	0.9919	0.9927	6.7

Table 3. The result with LINGO

Task 1	2nd S·P
Task 2	1st S·P
Task 3	2nd S·P
Task 4	1st S·P
Task 5	2nd S·P
Task 6	1st S·P
Task 7	2nd S·P
Task 8	2nd S·P
Task 9	
Task 10	
Task 11	2nd S·P
Task 12	

(a) Service Provider

	Customer's Requirement	Result
Duration	60s	59.7s
Cost	$ 800	$ 800.6
Reliability	95%	96.88%
Availability	95 %	95.35%
Reputation	8	7.9

(b) Results of LINGO

6 Conclusion and Future Work

Web Service can currently be recognized as a new alternative to overcome the conventional Internet business solutions. However, due to the increasing number of Web Service applications and the service suppliers, customers are confronted with the problem of selecting the most suitable web-service. In this paper, when selecting Web Service suppliers, Goal Programming can be used to guarantee the QoS of Web Service process. The proposed approach has several advantages compared with the other researches. First, the optimal Web Service suppliers can be selected with all QoS evaluation criteria which are quantitatively defined. Secondly, the parallel structures are also considered, claiming that the proposed approach can be applied more generally. Thirdly, this method can be applied at other domains like SCM or IT outsourcing when searching the partners and their QoS requirement. However, the dependencies between tasks are ignored in this paper. Also the QoS suggested by service suppliers are assumed to be always reliable. We will explore these issues more deeply by considering the real situation and the dynamic binding for Web Service selection in future work.

Acknowledgement

This work was supported by the Ministry of Science and Technology (MOST)/ Korea Science and Engineering Foundation (KOSEF) through the Advanced Information Technology Research Center (AITrc).

References

[1] Aalst, W.: Business Alignment: Using Process Mining as a Tool for Delta Analysis, In: Proc. CAiSE Workshops (2004) 138-145

[2] Basu A., Blanning, R.: Synthesis and Decomposition of Processes in Organizations, *Information Systems Research*, Vol. 14, No. 4 (2003) 337-355

[3] Cardoso, J., Sheth, A., Miller, J., Arnold, J., Kochut, K.: Quality of Service for Workflows and Web Service Process. *Journal of Web Semantics*, Vol. 1, Issue 3 (2004) 281-308

[4] Dogac, A., Laleci, G., Kabak, Y., Cingil, I.: Exploiting Web Service Semantics: Taxonomies vs. Ontologies. *IEEE Data Engineering Bulletin,* Vol. 25, No.4 (2002) 10-16

[5] Fensel, D., Bussler, C.: The WebService Modeling Framework WSMF. *Electronic Commerce Research and Applications,* Vol. 1, No. 2 (2002) 113-137

[6] Jaeger, M., Rojec-Goldmann, G., Mühl, G.: QoS Aggregation for Web Service Composition using Workflow Patterns. In: Proc. *EDOC* (2004)149-159

[7] Miller, G.: The Web Services Debate: .NET vs. J2EE. *Communications of the ACM*, Vol. 46, No. 6 (2003) 64-67

[8] W3C Web Services Architecture Working Group (http://www.w3.org/2002/ws/arch/)

[9] Williams, J.: The Web Services Debate: J2EE vs. .NET. *Communications of the ACM*, Vol. 46, No. 6 (2003) 58-63

[10] Yang, J., Papazoglou, M.: Service Components for Managing the Life-Cycle of Service Compositions. *Information Systems*, Vol. 29, No. 2 (2004) 97-125

[11] Zeng, L., Benatallah, B., Ngu, A., Dumas, M., Kalagnanam, J., Chang, H.: QoS-Aware Middleware for Web Services Composition. *IEEE Transactions on Software Engineering*, Vol. 30, No. 5 (2004) 311-327

[12] Martello S., Toth, P.: *Knapsack Problem: Algorithms and Computer Implementation*, John Wiley and Sons, 2001.

Healthcare Home Service System Based on Distributed Object Group Framework

Chang-Sun Shin[1], Chung-Sub Lee[2], and Su-Chong Joo[2]

[1] School of Information and Communication Engineering,
Sunchon National University, Korea
csshin@sunchon.ac.kr
[2] School of Electrical, Electronic and Information Engineering,
Wonkwang University, Korea
{cslee, scjoo}@wonkwang.ac.kr

Abstract. This paper suggests a healthcare home service system based on the Distributed Object Group Framework (DOGF) for ubiquitous healthcare in home environment. This system consists of 3 layers. The lower layer includes the physical sensors and devices for healthcare, as a physical layer. The middle layer is the DOGF layer. This framework supports the object grouping service and the real-time service to execute the healthcare application. Here, object group means the unit of logical grouped objects or healthcare sensors/devices for a healthcare service. We define these grouped objects as an application group, also sensors/devices as a sensor group. And this layer includes interfaces between application group at the upper layer and sensor group at the physical layer. The upper layer implements healthcare applications based on lower layers. With healthcare applications, we implemented the location tracking service, the health information service, and the titrating environment service. Our system can provide healthcare application services using the healthcare information from the physical healthcare sensors/devices, and also can be monitored and controlled the execution results of these services via remote desktops or PDAs.

1 Introduction

Modern computing environment has been changed into ubiquitous computing environment that shares information and services between the information systems and related devices/sensors. This computing paradigm influences to the whole human life style. There, in healthcare field, are many researches providing healthcare services by connecting the healthcare devices/sensors with home network [1]. Now we define this service, as the healthcare home service. The healthcare home service based on ubiquitous environment extends the existing healthcare services provided in medical institutions to the individual and the home [2, 3]. But, most of these systems have been constructed by using the dedicated system and application solutions to operate in the dependent environment like hospitals. That is, they cannot provide the integrating environment that can implement the new application services through the reusability/reconfiguration of the existing system's components or applications. Especially, the research for integrating the various sensors and applications to support

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 798 – 807, 2006.

total healthcare services are insufficient.

Hence, this paper, by adding and extending the functions of healthcare services to the Distributed Object Group Framework (DOGF) we developed [4, 5], suggests new system supporting the integration of sensors and applications providing healthcare service. Our system defines physical devices/sensors and applications supporting a healthcare service as a logical group. Also, for satisfying real-time constraints like emergency, we use the Time-triggered Message-triggered Object (TMO) scheme [6, 7] supporting real-time environment for implementing objects for healthcare services. Finally, by showing the example of the healthcare home service application in the system proposed, we verify that our system can provide various healthcare home services in ubiquitous home environment.

Section 2 presents the trends of healthcare service and the DOGF. Section 3 explains the structure and the supporting services of our healthcare home service system. In Section 4, we verify the execuability of the system via example of healthcare home services. And Section 5 discusses conclusions and future works.

2 Related Works

2.1 Technical Trends of Healthcare Home Service

The healthcare service in home extends the medical space to not only human's daily activity but also the life space including economy and culture activities. Therefore, the existing healthcare services, by integrating with other activities in life space, create new ubiquitous healthcare service that can overcome the limit of medical space [8, 9]. The researches for the healthcare home service with integrating infrastructure like home automation and home/sensor network have been studied in America, Europe, and Japan. The representative researches of healthcare home service are the Smart Medical Home suggested by Center of Future Health at University of Rochester and the eHill House by Matsushita in Japan. Figure 1 is showing the Smart Medical Home. The goal of this system is development of complete personal health system in home.

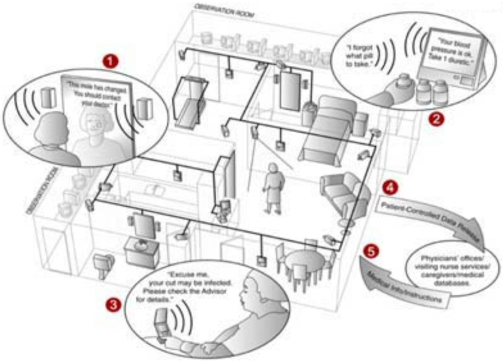

OBSERVATION ROOM
1
"This mole has changed. You should contact your doctor."
"I forgot what pill to take."
"Your blood pressure is ok. Take 1 diuretic."
2
OBSERVATION ROOM
4
Patient-Controlled Data Release
5
Physicians' offices/ visiting nurse services/ caregivers/medical databases
Medical Info/Instructions
"Excuse me, your cut may be infected. Please check the Advisor for details."
3

Fig. 1. Smart Medical Home

But, these systems depend on the sensing devices, the communicating and processing modules, and other hardware. Due to above problems, the systems are difficult to add new healthcare sensor and application, and provide integrating environment. For overcoming the system dependent service environment, this paper proposes the healthcare home service system based on the DOGF that can reuse and reconfigure the healthcare sensors/devices and the healthcare applications in home.

2.2 Distributed Object Group Framework

We extend the DOGF that we suggested [4,5] to develop the new healthcare home service that can provide dynamic binding service between them and grouping service the various sensors and the applications related with healthcare home service. This framework supports a logical single view system environment by grouping them. Our framework also provides the distributed transparency for complicated interfaces among distributed objects existing in physical distributed system, locates between Commercial-Off-The-Shelf (COTS) middleware layer and distributed application layer. It is main functional components that consist of object group management supporting component and real-time service supporting component. For supporting the group management service, our framework includes the Group Manager (GM) object, the Security object, the Information Repository object, and the Dynamic Binder object with server objects. For real-time service, the Real-Time Manager (RTM) objects and Scheduler objects exist in the framework. Figure 2 is showing the structure of the DOGF.

This paper considers the TMOs configuring healthcare application as an application service group by extending the DOGF, and proposes the healthcare home service system that can monitor health status for residents and control home environment to provide residents with the appropriate life environment.

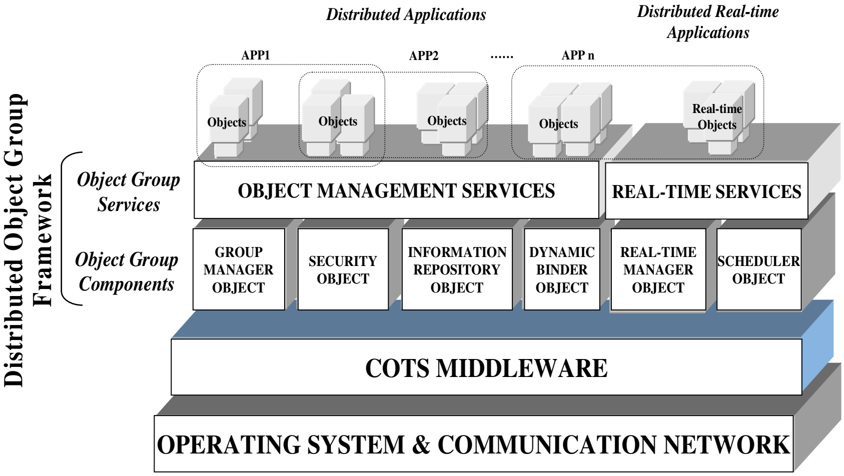


Fig. 2. Architecture of Distributed Object Group Framework

3 Healthcare Home Service System

In this Section, we develop the healthcare home service system that supports the resident's healthcare in their home by extending the DOGF. This system supports the grouping of distributed applications, devices, and sensors for healthcare service by interacting between the components of the DOGF.

3.1 Structure of Healthcare Home Service System

Our system classifies the existing healthcare services in home into the location tracking service, the health information management service, and the titrating environment service. These services support the healthcare home service by reconfiguring or grouping them as application groups. The DOGF on the middle layer supports the execution of application of appropriate healthcare home service on the upper layer by using the input information obtained from the individual or grouped physical devices on the lower layer. That is, according to the services or status of the home network for healthcare, our system could reconfigure new healthcare services dynamically by integrating physical healthcare devices/sensors on the lower layer and healthcare application on the upper layer, horizontally or vertically. Figure 3 is showing the structure of the healthcare home service system based on the DOGF.

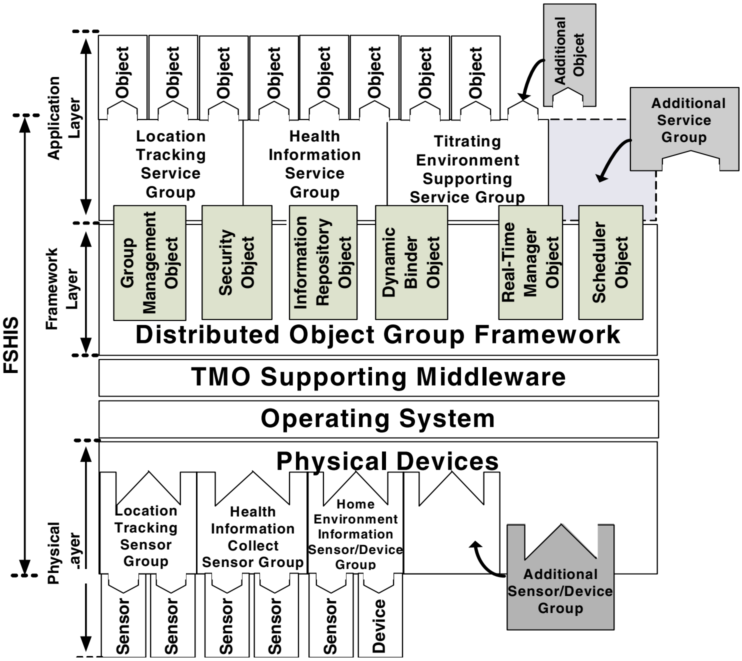

FSHIS
Application Layer
Framework Layer
Physical Layer
Object
Object
Object
Object
Object
Object
Object
Object
Additional Objcet
Additional Service Group
Location Tracking Service Group
Health Information Service Group
Titrating Environment Supporting Service Group
Group Management Object
Security Object
Information Repository Object
Dynamic Binder Object
Real-Time Manager Object
Scheduler Object
Distributed Object Group Framework
TMO Supporting Middleware
Operating System
Physical Devices
Location Tracking Sensor Group
Health Information Collect Sensor Group
Home Environment Information Sensor/Device Group
Additional Sensor/Device Group
Sensor
Sensor
Sensor
Sensor
Sensor
Device

Fig. 3. Structure of Healthcare Home Service System Based on the DOGF

For satisfying the requirements of the healthcare home service from the structure of the DOGF, we redefine the interactions among the distributed service objects, the various physical devices, and the group management components. We don't consider the real-time supporting components in the DOGF due to using the TMO scheme supporting real-time property itself and the TMOSM, distributed real-time middleware. When the new healthcare home service application is required, our system could add or delete the physical devices, the application service groups, and distributed objects as shown in Figure 3. Table 1 is showing the distributed services provided by the components in existing DOGF, and the supporting services of healthcare home service system extending the framework.

Table 1. Supporting Services of Hralthcare Home Service System

Component of the DOGF	Supporting services of the DOGF	Supporting services of the healthcare home service system
-GM object -Information Repository object	-Object group supporting service	-Grouping service of the healthcare supporting devices/sensors -Object grouping service of the healthcare supporting distributed objects
-GM object -Security object	-Access right control service	-Access right control service based on properties of healthcare data, devices, sensors, and application groups.
-GM object -Dynamic Binder object	-Dynamic object selection and binding service	-Dynamic binding service for duplicated healthcare resources

3.2 Supporting Services of the Healthcare Home Service System

The healthcare home service system provides the location tracking service, the health information service, and the titrating environment service through logical grouping of physical devices, sensors, and distributed applications for supporting home healthcare as showing in Figure 4.

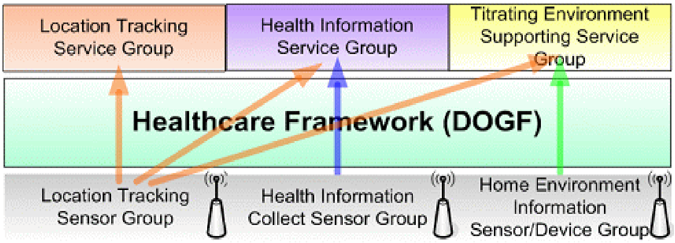

Location Tracking Service Group
Health Information Service Group
Titrating Environment Supporting Service Group
Healthcare Framework (DOGF)
Location Tracking Sensor Group
Health Information Collect Sensor Group
Home Environment Information Sensor/Device Group

Fig. 4. Executing Environment between Sensor Groups and Service Groups for Healthcare

Location Tracking Service. The location tracking service concerned in our paper gives the location information of the moving resident obtained from many sensor nodes with the functionalities of sensing, information processing, and communication. We use the Cricket sensor node developed in MIT [10]. For providing the location tracking service, the DOGF logically define the location tracking service group by grouping the locating sensor and the TMO application objects. With the location tracking service, we could not only observe the moving resident in the home area periodically, but also have the real-time monitoring about the time length of staying in an individual space, the moving distance per hour, the current position of the resident, and so on.

Health Information Service. The health information service could be executed through the grouping of the health information service group on the upper layer, the health information sensor group and the location tracking sensor group on the physical. For example, we define the healthcare information sensors that are temperature, electrocardiogram, blood pressure, and glycosuria sensors and then collect the location information of resident via the location tracking sensor group mentioned above. The DOGF is responsible to integrate above groups. Our system provides the management service of healthcare information and emergent status based on health information service group, location tracking sensor group, and health information sensor group.

Titrating Environment Service. The titrating environment service is executed by grouping the titrating environment supporting service group on the upper layer and the physical devices such as the home environment information sensor/device group and the location tracking sensor group on the physical layer. In this service, we use information appliances, indoor temperature sensor, illumination sensor, and humidity sensor as physical devices, and define TMOs for application service groups. The service provides not only the real-time control service that can control the activity property of information appliances by changing strong/medium/weak power, but also the adapting environment controlling service for matching into client's request. These can maintain the appropriate indoor temperature, illumination, and humidity through the real-time monitoring and controlling of individual home appliances.

4 Implementation of Healthcare Home Service System

This Section describes the definition of the executing objects that are each service group's components of distributed application implemented in our suggested healthcare home service system. Also, for providing healthcare home services, we implement the sensor group on the lower layer and the executing object group on the application layer, and then show the remote monitoring results in integration environment. For implementing the system, we use the TMO scheme and the TMOSM developed by DREAM Lab. at University of California at Irvine [6, 7]. The TMO has the Service Method (SvM) triggered by client's request and the Spontaneous Method (SpM) that can be spontaneously triggered by the defined time in an object by extending the execution characteristics of existing object. This object interacts with others by remote calling. In our system, the objects configuring healthcare home service are implemented by TMO scheme.

4.1 Definition of Service Components

The components of the healthcare home service system are defined by the TMO scheme. First, for the location tracking service, the Person TMO is mapping to moving object, called resident, in simulation environment, and sensed by physical sensor (Cricket). The Sensor TMO senses the moving resident by the periodic time description, stores the location information of Person TMO into information repository, ODS. When detecting the moving object, Sensor TMO transfers the location information to the Location Tracking TMO. And then, the information transfers to the Monitor TMO. We check the visiting counts and the current location by using this information. Monitor TMO reflects the related service and location information of resident into the 2 dimensional simulation spaces. When Person TMO doesn't move for the specified period, Emergency TMO sends an urgent request to the particular hospital after notifying the 1-step emergency to the corresponding home. The Tonometer TMO periodically sends the blood pressure information of the resident to the Location Tracking TMO. And the Glycosuria TMO transfers the glycosuria result obtained from the glycosuria sensor installed in lavatory to the Location Tracking TMO. The Location Tracking TMO provides the health information to the Monitor TMO. The Home Server TMO, component for the titrating environment service, monitors the action of all information appliances by receiving the information from corresponding appliance's TMO. The Air conditioner TMO, the Heater TMO, and the Fan TMO control the indoor temperature. The Light TMO controls the illumination in home. The Camera TMO observes a thief at nighttime. The Window TMO changes the indoor air condition periodically. Also, the Humidity TMO notifies the indoor humidity information to the Home Server TMO. Figure 5 describes the interaction with the components for healthcare home service in our system.

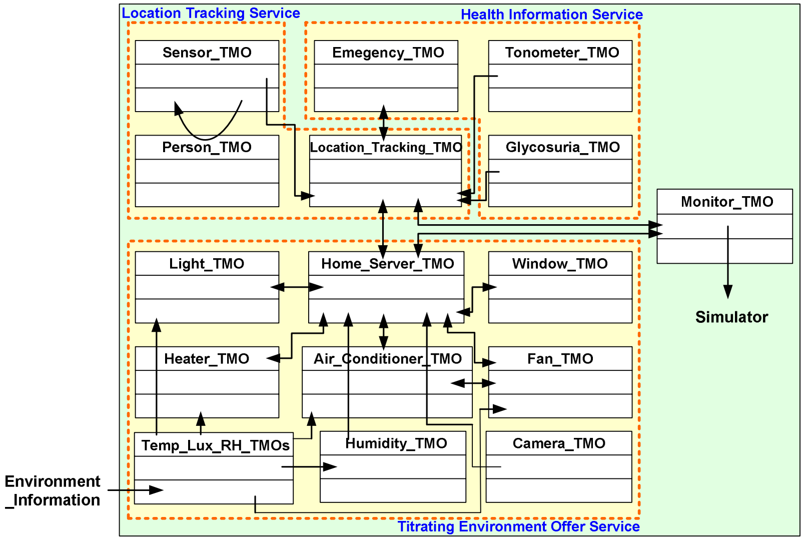


Fig. 5. Interaction of Applications, TMOs, supporting the Healthcare Home Service

4.2 Execution Conditions of TMO for Each Healthcare Service

Table 2 defines the execution conditions of TMOs configuring the healthcare home service applications. Now, ON and OFF mean the start and the stop time which sense

Table 2. Execution Conditions of TMOs supporting the Healthcare Home Service

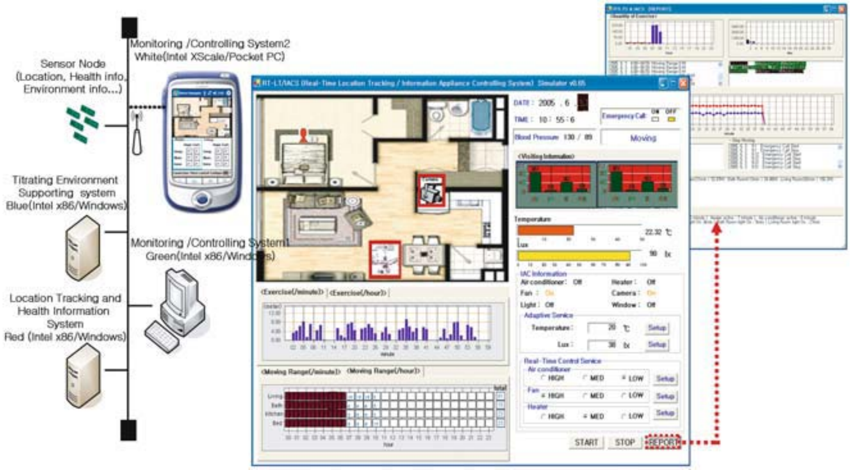
Supporting service	Executing TMOs	Execution conditions	
Location Information Service (Location Information)	Sensor TMO	Location tracking and seeking of Home resident	
	Person TMO		
	Location Tracking TMO		
Health Information Service (Blood pressure, glycosuria, and time)	Tonometer TMO	ON	Blood pressure is over 150/95 Blood pressure is under100/70
	Glycosuria TMO	ON	Under 70mg/dl or Over 130mg/dl
	Emergency TMO	ON	When doesn't move for over 10 minutes
Titrating Environment Service (Temperature, illumination, and humidity)	Air Conditioner TMO	ON	Temperature is over 27℃
		OFF	Temperature is under 23℃
	Fan TMO	ON	Temperature is between 25℃ and 27℃
		OFF	Temperature is under 20℃ or over 27℃
	Heater TMO	ON	Temperature is under 12℃
		OFF	Temperature is over 18℃
	Light TMO	ON	Illumination is under 40*lx*
	Camera TMO	ON	According to the setting time
	Window TMO	ON	For 5 minutes per 30minutes

the execution situation of each TMO according to the execution conditions like resident location, blood pressure, home temperature, illumination, and time. The execution conditions define at the Autonomous Activation Condition (AAC) in TMO's SpM, and the TMO acts by referring the AAC. If the given execution conditions are satisfied, each TMO can execute spontaneously and collect the execution condition's value, which are the moving distance, blood pressure, glycosuria, temperature, illumination, humidity, and time, from the physical sensors.

4.3 Execution Results of Healthcare Home Service System

We designed the healthcare home service system by grouping the physical sensors and the TMOs providing healthcare service through the components supporting the group management service in the DOGF. To satisfy the real-time requirements of healthcare services, we implemented each service object based on TMO scheme. This system provides the healthcare home service based on the DOGF to integrate services into logical groups. Figure 6 shows the physical environment and the execution results of the healthcare home service system. The components for the location tracking service group and the health information service group are installed on one system (system name is Red in Fig. 6). And, the component for the titrating environment service is installed on the other system (called Blue). The GUI system monitoring and controlling the healthcare home service is located on the desktop system (called Green) and PDA called White. From the GUI in Figure 6, we could see the execution results of 3 healthcare supporting services mentioned in Section 3. The monitoring and controlling information are collected real-time while executing the healthcare home service. And, we construct the healthcare database to analyze statistics easily by using the information.

Figure 7 describes a Healthcare Home Model reflecting real world. This model interacts with the executing environment shown in Figure 6. From the model, we verify the executbility of the healthcare home service system we developed.


Sensor Node
(Location, Health info,
Environment info...)
Titrating Environment
Supporting system
Blue(Intel x86/Windows)
Location Tracking and
Health Information
System
Red (Intel x86/Windows)
Monitoring /Controlling System2
White(Intel XScale/Pocket PC)
Monitoring /Controlling System1
Green(Intel x86/Windows)
RT-LT/IACS (Real-Time Location Tracking / Information Appliance Controlling System) Simulator v0.65
<Exercise(/minute)> <Exercise(/hour)>
<Moving Range(/minute)> <Moving Range(/hour)>
Living
Bath
Kitchen
Bed
total
RT-LT/IACS [REPORT]

DATE : 2005 . 6

TIME : 10 : 55 : 6

Emergency Call: ON □ OFF ■

Blood Pressure: 130 / 89

Moving

<Visiting Information>

Temperature 22.32 ℃

Lux 90 lx

IAC Information

Air conditioner : Off Heater : Off

Fan : On Camera : On

Light : Off Window : Off

Adaptive Service

Temperature : 20 ℃ Setup

Lux : 38 lx Setup

Real-Time Control Service

Air conditioner ○ HIGH ○ MED ● LOW Setup

Fan ● HIGH ○ MED ○ LOW Setup

Heater ○ HIGH ● MED ○ LOW Setup

START STOP REPORT

Fig. 6. Physical Environments and Execution Results of Healthcare Home Service System

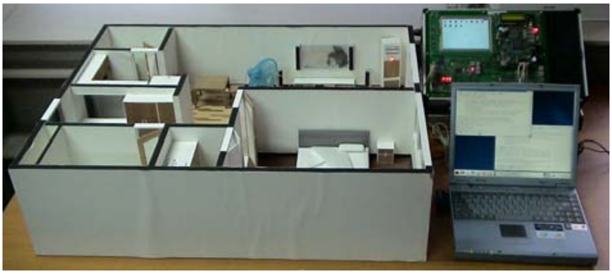


Fig. 7. Healthcare Home Model supporting Healthcare Services

5 Conclusions and Future Works

In this paper, we suggested the Healthcare Home Service System based on the Distributed Object Group Framework (DOGF) for supporting the ubiquitous healthcare services on home network. This system consists of 3 layers. The lower layer includes the physical sensors and devices for healthcare, as a physical layer. The middle layer is the DOGF layer. And the upper layer implements healthcare applications based on lower layers. We also implemented the location tracking service, the health information service, and the titrating environment service. At this time, our system could support the functional grouping of each application for the healthcare home service and the dynamic binding among these groups by using the DOGF. That is, we defined physical sensors/devices and distributed objects supporting healthcare as the logical single sensor group and the distributed application service group, respectively. And, by defining the interfaces between the sensor group and the application service group, we could develop the additional healthcare services by creating new service group through the interactions of the TMOs which are the implementing objects executing each service.

In the future, we will develop the mobile agent or proxy for the location based application services in the DOGF layer. Then we are to apply the various healthcare services to this system, and verify the executability of our system by comparing and analyzing with the existing systems.

Acknowledgements. This work was supported by the Regional Research Centers Program of the Korean Ministry of Education & Human Resources Development through the Center for Healthcare Technology Development.

References

1. Seung-Chul Shin and et al: An implementation of e-Health System for sensing emergent conditions. Journal of the Korea Information Science Society, Vol. 31, No. 1 (2004) 322–324
2. Berler, A., Pavlopoulos, S, Koutsouris D: Design of an interoperability framework in a regional healthcare system. In Proceedings of Engineering in Medicine and Biology Society, Vol. 2 (2004) 3093–3096
3. K. Seshadri, L. Liotta, and R. Gopal, T. Liotta: A wireless Internet application for healthcare. In Proceedings of 14th IEEE Symposium on Computer-Based Medical Systems (2001) 109–114
4. Chang-Sun Shin, Chang-Won Jeong, Su-Chong Joo: TMO-Based Object Group Framework for Supporting Distributed Object Management and Real-Time Services. Lecture Notes in Computer Science, Vol. 2834, Springer-Verlag, Berlin Heidelberg and New York (2003) 525–535
5. Chang-Sun Shin, Chang-Won Jeong, Su-Chong Joo: Construction of Distributed Object Group Framework and Its Execution Analysis Using Distributed Application Simulation. Lecture Notes in Computer Science, Vol. 3207, Springer-Verlag, Berlin Heidelberg and New York (2004) 724–733
6. Kim, K.H., IShida, M., and Liu, J.: An Efficient Middleware Architecture Supporting Time-triggered Message-triggered Objects and an NT-based Implementation. In Proceedings of the IEEE CS 2nd International Symposium on Object-oriented Real-time distributed Computing (ISORC'99) (1999) 54–63
7. Kim, K.H: The Distributed Time-Triggered Simulation Scheme : Core Principles and Supporting Execution Engine. The International Journal of Time-Critical Computing Systems- Real-Time Systems, Vol. 26, No. 1 (2004) 9–28
8. S. Helal, B. Winkler, C. Lee, Y. Kaddoura, C. Giraldo, S. Kuchibhotla, W. Mann: Enabling Location-Aware Pervasive Computing Applications for the Elderly. In Proceedings of the First IEEE International Conference on Pervasive Computing and Communications (2003)
9. Heun-Kyung Lee: A Survey for Ubiquitous Computing Projects. The Journal of ETRI (2003)
10. The Cricket Indoor Location System. http://cricket.csail.mit.edu/ (2005)

A Learning Attitude Evaluation System for Learning Concentration on Distance Education

Byungdo Choi and Chonggun Kim*

Dept. of Computer Eng., Yeungnam Univ.,
214-1, Deadong, Kyongsan, Kyungbuk, 712-749 Korea
xenon@korea.com, cgkim@yu.ac.kr

Abstract. Due to the needs of the Internet applications, variety of distance education methods on Internet are provided. As the result, various multimedia contents for education have been developing. One problem of the distance education on the is to lead the students to concentrate to learning for improving good attitude. But it is quite difficult to evaluate the students' behaviors and participating sincerity on the distance lectures. In this paper, we propose an evaluation method of the learning attitude and a practical system in a Web-flash based distance education environment. In the proposed system, students of distance education are evaluated by tracking their behaviors, and thus their sincerity can be estimated as well.

1 Introduction

With the development of multimedia applications as well as the wide use of the Internet, the distance education is rapidly spreading and being applied to various fields of study. Currently, many institutes and universities have incorporated with distance education systems, and the systems are at a stage of continuous development.

The distance education on the Internet, a consumer-oriented education, give the easier access to the various multimedia than the traditional education. In the distance education, the most important element is of course the contents of the education[1]. The lecture contents may be created using web pages, multimedia creation tools, real media or animations[2], [3], [4]. Another important issue is how to delivery the contents to students effectively. In order to increase the effectiveness of the lectures, realtime discussions or level-based courseware are used. To increase studying performance, students have to concentrate for using the contents, but the only method of evaluating the students is through test scores or report evaluations. There is no way of valuating the fulfillment degree of the lecture content and the degree of participation in class [5], [6], [7], [8].

In this paper, we propose a student attitude evaluation system in an Web-flash based distance education contents. Through this system, the students' behaviors can be

* Correspondence author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 808 – 817, 2006.

tracked and accordingly provided with proper feedback and guidance which may be able to stimulate the students' study motivations. This system can be useful when study attitude evaluating results are considered as a proper factor for grading students in many educational institutes.

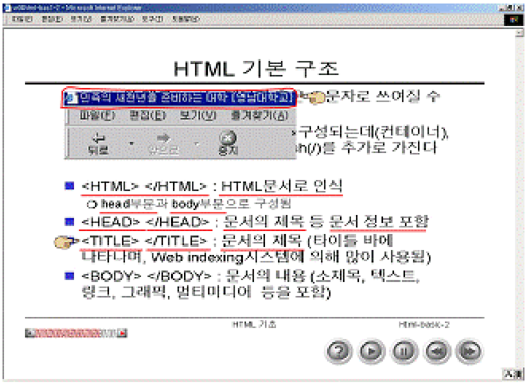
2 Related Works

2.1 Mastery Learning

The mastery learning theory[9] is an educational theory that criticizes the intelligence theory[10] and states that the students, by investing adequate amount of time, will be able to understand 90% of the materials covered in a well organized class. Therefore, the learning process may be different for each individual student. In the mastery learning theory, the student can move on to the next study unit only if the student passes the test given at the end of each unit. The mastery learning theory enables the students to understand better the lessons learned before moving on to the next study unit, and the teachers to assess better the students' learning process.

2.2 A Cell and a Handover Area Model

An Web-flash based lecture contents is a method that uses animation, in which the instructor creates the lecture specifications using the lecture creation tools. With the newly created lecture specifications, the instructor manually adds various effects that might aid the students' understanding of the lecture material, and completes the creation of other effects. In comparison to other creation tools or distance lecture contents, the Web-flash based contents provide the following advantages. Because the Web-flash based contents are also based on HTML(Hypertext Makeup Language), internet traffic can be reduced and students can take the lectures on any computer only with a web browser. Also, the contents can be made to be dynamic, which makes students are less bored when taking the lecture. Another advantage of the Web-flash based contents is that these contents can be easily modified or updated. A drawback



HTML 기본 구조

문자로 쓰여질 수

구성되는데(컨테이너),
h(/)를 추가로 가진다

■ <HTML> </HTML> : HTML문서로 인식
○ head부문과 body부문으로 구성됨
■ <HEAD> </HEAD> : 문서의 제목 등 문서 정보 포함
<TITLE> </TITLE> : 문서의 제목 (타이틀 바에 나타나며, Web indexing시스템에 의해 많이 사용됨)
■ <BODY> </BODY> : 문서의 내용 (소제목, 텍스트, 링크, 그래픽, 멀티미디어 등을 포함)

HTML 기초 Html-basic-2

Fig. 1. Web-flash based the distance lecture example

of the Web-flash based might be the fact that the instructor must go through two stages of preparation when creating the lectures. The following fig. 1 is an example of the Web-flash based the distance lecture that is currently being offered at Yeungnam University [11].

3 Tracking System Design

In order to evaluate the students' sincerity, the instructor must attain various information concerning the behavior of the learner, and can evaluate the students' behavior status based on this information an evaluation. In this paper, an instructor should consider the follows for proper evaluation : how much time they study the content, how much they understand the content, how much they fulfill their learning schedules, and whether they do steadily and diligently. All studying behavior of the students are recorded through the tracking system and these information are stored in a DB server through a web server. The following fig. 2 shows the concept of tracking system.

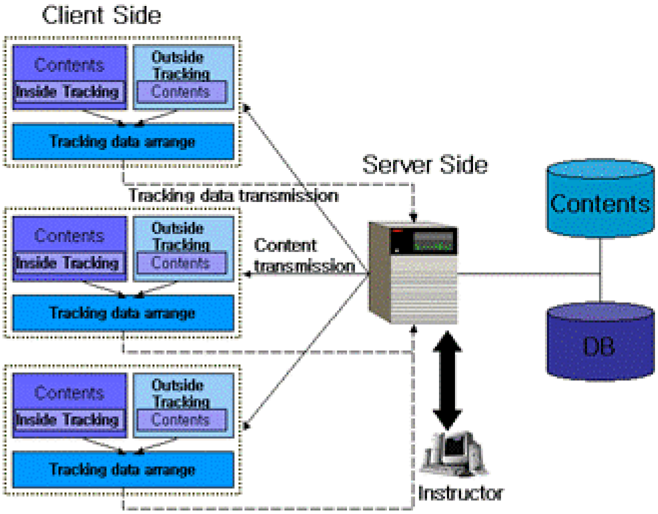


Fig. 2. Concept of tracking system

This student attitude tracking system has been implemented in two different methods. The internal tracking data is used in student evaluations and the external tracking data is used to analyze student patterns for the future.

3.1 Internal Tracking

Internal tracking is a method that uses Flash Action Script in which the different actions and corresponding variable values are inserted into the buttons on each slide of the lecture. When an action occurs on a corresponding button, the student id, action type and time, frame information and the slide number are transmitted to the web server. With the transmitted information, the instructor is able to understand better the current learning process of the student. Fig. 3 shows the structure of the internal tracking.

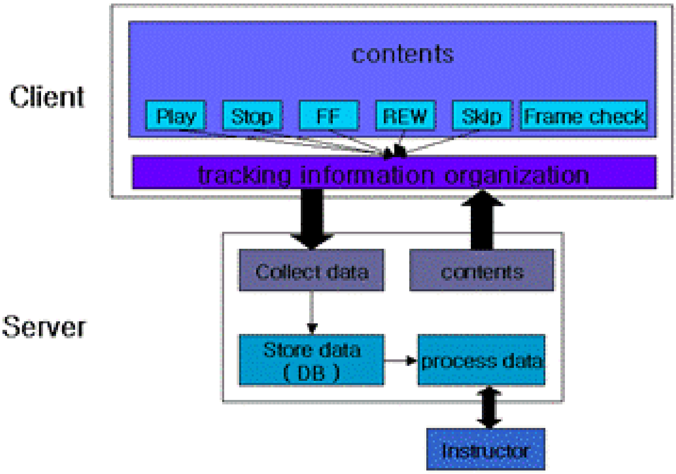


Fig. 3. Internal tracking structure

3.2 External Tracking

Instead of the Flash Player on an web browser, the external tracking uses Flash tracker, a Flash Tracker that is created with Visual Basic's Active X component, to

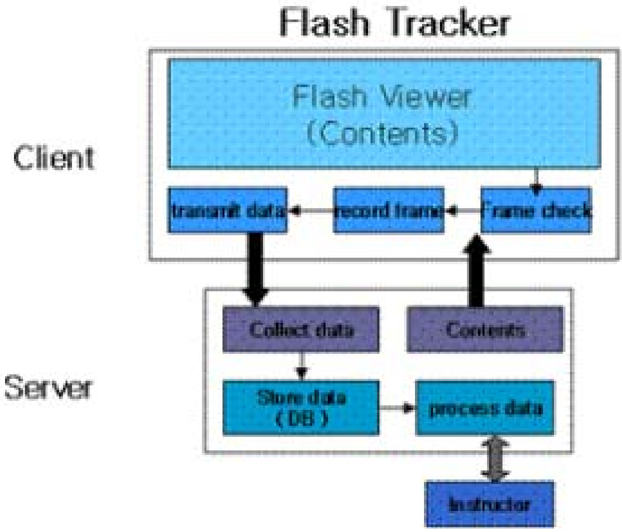


Fig. 4. Flash tracker structure

display the lecture contents and to track learning information. While the student is taking the lecture, the content frame information are continuously recorded and this information is temporarily stored onto the computer memory before being transmitted to a designated DB server when the lecture is over. The following fig. 4 shows the structure of the Flash tracker.

The Flash tracker replaces the CLSID(Class ID) and the CODEBASE of the OBJECT tag within the existing html lecture file with the Flash tracker information.

3.3 Learning Process

The following fig. 5 is the flow chart showing the overall learning process.

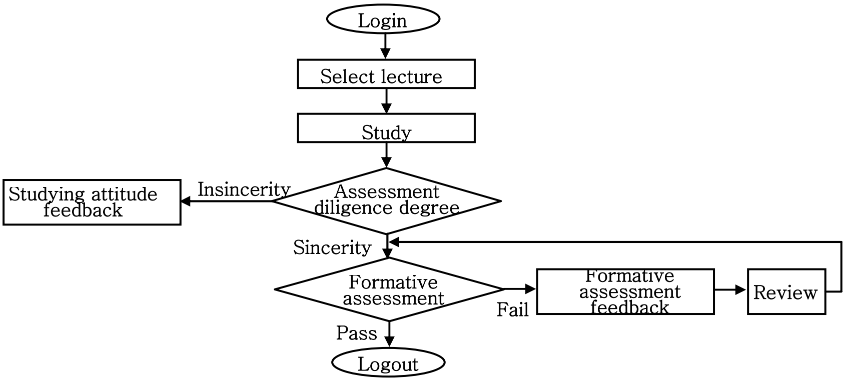


Fig. 5. The overall learning process for evaluation

A student logins and selects the contents in order to begin the study. The student's study activities start to be recorded through the tracking system at the same time the study begins, and this information is stored in a DB server. When the student finishes the lecture, then the data stored in the DB is analyzed, and if the analysis results satisfy sincerity and formative evaluation conditions, the lecture is completed, however if the conditions are not satisfied, the student has to retake the lecture.

The following fig. 6 shows the tracking of a study unit. Each study unit represents a slide in the lecture and the data to evaluate the student sincerity is collected according to each slide. The data gathered is used to evaluate the student sincerity for each slide at the end of the study.

3.4 Evaluation Algorithm

The student evaluation is made based on the students' study behavior and the mastery learning theory. This evaluation therefore centers upon the students' sincerity in their studies. A student will be considered to have completed the study with good sincerity if the student information gathered from the internal tracking is showed as table 1.

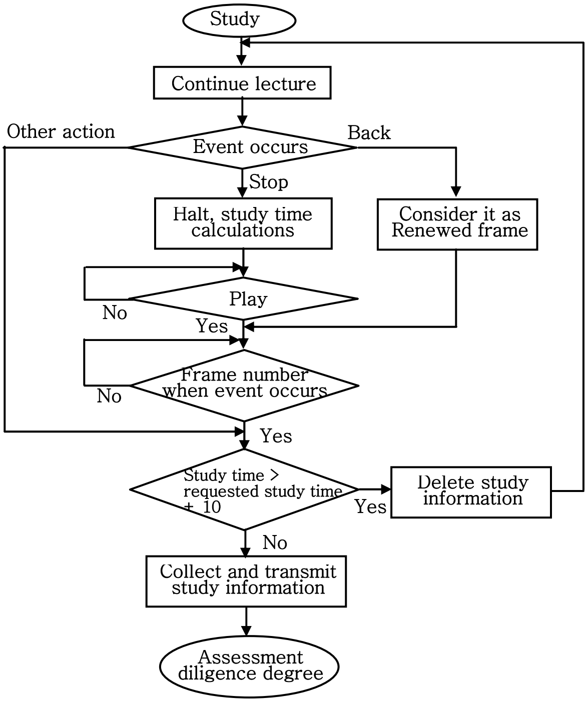

Study
Continue lecture
Other action
Event occurs
Back
Stop
Halt, study time calculations
Consider it as Renewed frame
Play
No
Yes
Frame number when event occurs
No
Yes
Study time > requested study time + 10
Yes
Delete study information
No
Collect and transmit study information
Assessment diligence degree

Fig. 6. Tracking of a study unit

Table 1. Sincere student data

	Tracking data
Study time	Taken at least 90% of slide lectures
Learning process	At least 90% lecture process
Attendance	Taken at least 90% of the lectures without skipping
Formative assessment	Assessment score of 90 or greater

The study time is the actual time that the student has taken the lecture, that is, the time from the beginning of the lecture to the time the student went on to the next unit, or to the time that the student has completed the lecture. The equation (1) is the study time.

T_1 : study time, T_p : process time, T_r : review time, T_s : stop time, T_e : time after completion

$$T_1 = T_p - (T_r + T_s + T_e). \tag{1}$$

The learning process status is the largest frame number among the frames that have been transmitted to the server since the beginning of the study to the end. The equation (2) is the learning process status.

F_l : learning process, F : frame number

$$F_l = \text{MAX}\ \{F \mid \text{Frame Number}\}. \tag{2}$$

The attendance status shows whether the student has taken all the lectures. This is determined by the check points on certain frame position information in whose values are transmitted to the server where the number of check points are compared. The equation (3) is the attendance status.

C_l : learning check points, C_c : collected check points, C_r : number of review check points

$$C_l = C_c - C_r \tag{3}$$

Formative evaluation determines whether the student has adequately understood the lecture by comparing the acquired scores. In our case, the student must take at least 90% of the lectures in order to understand the materials covered. Therefore, the student sincerity evaluation is made by using the study time rate, learning process rate, and attendance rate by the following equations.

T_{tl} : required study time, F_{tf} : last frame number, C_{tl} : total number of check points

$$study\ \ time\ \ rate : R_t = \frac{T_l}{T_{tl}} \times 100(\%). \tag{4}$$

$$learning\ \ process\ \ rate : R_f = \frac{F_l}{F_{tl}} \times 100(\%). \tag{5}$$

$$attendance\ \ rate : R_c = \frac{C_l}{C_{tl}} \times 100(\%). \tag{6}$$

```
For($i=0; $i <entire slide; $i++) {
        If ($R_t[$i] >= 0.9){
                If ($R_f[$i] >= 0.9){
                        If ($R_c[$i] >= 0.9){
                                $master[$i]=1;
                        }else{ $master[$i]=0;
        } } }
        If($master[$i] == 1){
                $count = $count +1;
        }else{ echo("please study slide $i again");
} }
if($count == total number of slides){
        if($ans_count >= number of problems*0.9) {
                echo("you have sincerely completed the lecture.");
        }else{ echo("you have not adequately understood the lecture.");

  } }
```

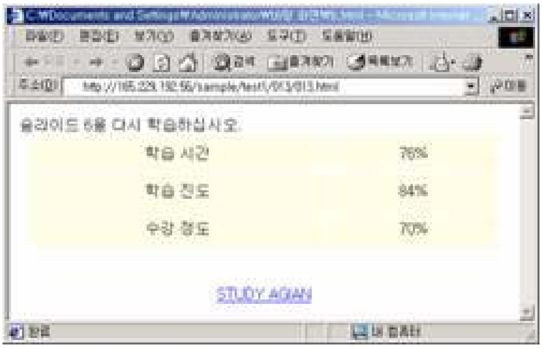
Fig. 7. The evaluating algorithm for sincerity and formative evaluation

We decide as an exemplary sincerity studying model when all the R_t, R_f, R_c have over 90%. The algorithm for evaluating the student achievement using student sincerity and formative evaluation is shown on fig. 7.

4 Results of Implementation

We implement a prototype system on a university class contents in order to test the tracking system. We take results from the student evaluation which have been determined using the internal tracking data.

Fig. 8 shows a feedback slide in which the student did not study sincerely, and the sincerity percentage according to the each category is displayed as studying time rate: 76%, learning process rate: 84%, attendance rate: 70%. By displaying the categories in which the student did not study sincerely, the student will attend to the lecture with better sincerity.


파일(F) 편집(E) 보기(V) 즐겨찾기(A) 도구(T) 도움말(H)
검색
즐겨찾기
미디어
주소(D)
http://165.229.192.56/sample/test1/013/013.html
이동
완료
내 컴퓨터

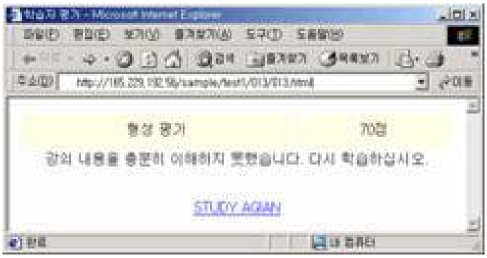
슬라이드 6을 다시 학습하십시오.

학습 시간	76%
학습 진도	84%
수강 정도	70%

STUDY AGIAN

Fig. 8. An insincere study case

Fig. 9 shows a student who has studied with sincerity but has not adequately understood the material covered. The message shows that the formative evaluation is 70. Students who have studied with sincerity will move to the formative evaluation level. If the formative evaluation score is below a certain standard, the formative


학습자 평가 - Microsoft Internet Explorer
파일(F) 편집(E) 보기(V) 즐겨찾기(A) 도구(T) 도움말(H)
검색
즐겨찾기
미디어
주소(D)
http://165.229.192.56/sample/test1/013/013.html
이동
완료
내 컴퓨터

형성 평가	70점

강의 내용을 충분히 이해하지 못했습니다. 다시 학습하십시오.

STUDY AGIAN

Fig. 9. Insufficient Formative Assessment Score

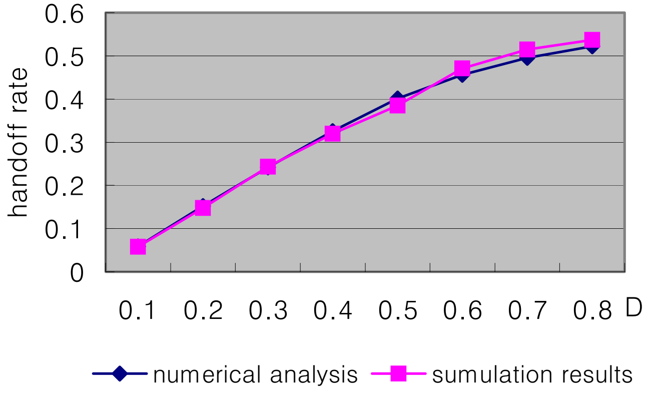


Fig. 10. Sincerity comparison

evaluation results are shown to the student and the student must retake the lecture in order to sufficiently understand the material covered.

The tracking method proposed in this paper has been applied to distance education at Yeungnam University and student evaluations have been made twice. The following Fig. 10 shows the student sincerity in the case where the evaluation results from tracking the student's academic activities are feedbacked to the student and in the case where the evaluation results are not feedbacked.

In the case where the results from the student sincerity evaluation are feedbacked to the student, the overall sincerity average increased by approximately 20%.

5 Conclusions

In this paper, a learning attitude evaluation system for learners based on Web-flash multimedia education contents is proposed and implemented in a distance education environment. The system is able to evaluate the learner's learning achievement through analysis of the students learning process by tracking student's attitude. It can decide not just diligence degree by the use of data that is derived from inside tracking of contents but also learning achievement. This system that tracks the learners' behavior and evaluates their achievement after taking the lecture can give more detail feedbacks to each of the learners than ever before. In addition, using a decision of diligence degree it can stir them to participate in the lecture positively in the future. The analysis results can be applied for studying of learning patterns or the complementary works for upgrading contents afterwards.

References

1. Keongin Won "Design Strategy of Web Contents for Intensifying Interactivity on Online Remote Fducation", Journal of Korean Society of Communication Design, 2001.12. Vol. 04
2. Heinich, R.et.al, "Instructional Media and Technologies for Learning", Prentice-Hall, Inc., 1996.

3. Yongjun Choi, Jahyo Ku, Intaek Leem, Byungdo Choi, Chonggun Kim, “A Study On Distributed Remote Lecture Contents for QoS Guarantee Streaming Service”, Journal of Korea Information Processing Society, 2002, vol.9, No.4
4. Jaeil Kim, Sangjoon Jung, Yongjun Choi, Seongkwon Cheon, and Chonggun Kim, “Design and Implementation of Lecture Authoring Tool based on Multimedia Component”, Journal of Korea Multimedia Society, 2000, Vol.3, No.5
5. RuiMin Shen, YiYang Tang, TongZhen Zhang, “The Intelligent Assessement System in Web_Based Distance Learning Education”, IEEE 2001
6. Flora Chia-I Chang, "Intelligent assessment of distance learning", Information Sciences 2002
7. Taeseog Kim, Jonghee Lee, Keunwang Lee, Haeseok Oh, “Design of Multi-agent System for Course Scheduling of Learner-Oriented using Weakness Analysis Algorithm”, Journal of Korea Information Processing Society, 2001.12, Vol.A, No.8
8. Kyungho Choi, “The Design and Implementation of Learner-Analyzing System in the web-based Distance Education”, Journal of Korean association information of education, 2001, Vol.5, No.1
9. Bloom, B. S. (1968). Learning for mastery. Evaluation Comment, Vol.I, Los Angeles: Center fro the Study of Evaluation of Instructional Programs, University of California, Los Angeles.
10. Binet, A. , & Simon, T. (1905). Methods nouvelles pour le diagnostic du niveau intellectuel des anormaux. Annee Psycholique, 11, 191-224.
11. Chonggun Kim, Seung Pil Jung “The Method of Develop Multimedia Contents on Technology Subject in Distance Education”, Korea Multimedia Society Review, 2001.11, Vol.5, No.4

A Calculation Method for Direction Based Handover Rate in Cell Based Mobile Networks

Mary Wu and Chonggun Kim*

Dept. of Computer Eng., Yeungnam Univ.,
214-1, Deadong, Kyongsan, Kyungbuk, 712-749 Korea
mrwu@yumail.ac.kr, cgkim@yu.ac.kr

Abstract. WiBro(Wireless Broadband) is an emerging technology for portable Internet to support high speed rate. The communication infrastructure is the cellular system and is designed to maintain communication connectivity of mobile terminal at speeds of up to 60 km/h. The WiBro provides handover to support seamless communication. When the signal strength, coming from neighbor BS, is stronger than the threshold, the handover procedure is initiated. But, it can't be predicted whether the MS will really go out of the current cell or not.

If the moving direction and speed of MSs are considered, it is possible to discriminate unnecessary efforts for handover connections. In this paper, we propose an efficient direction and speed based handover connection control schemes for increasing the utilization of channels and reduce a probability of new connection blocking rate. Some results of computer simulation are evaluated to show the effectiveness of the proposed method.

1 Introduction

The WiBro, a high-speed mobile wireless Internet service, is an emerging technology which is developed in South Korea[1,2]. The WiBro operates at the 2.3 GHz broadband and the communication infrastructure is a cellular system. It is based on IEEE 802.16e standard and is designed to maintain connectivity on mobile environment at speed of up to 60 km/h. The WiBro provides handover to support the mobility of end users. It guarantees the continuity of the wireless services when the mobile user moves across the cell boundaries. The WiBro provides the soft handover. An MS can communicate with a new base station(BS) before interrupting the communication with the current base station. Therefore, the soft handover provides seamless communication to the MS. Since the failure of a handover results in forced interruption of sessions, it derives a lot of data missing in high-speed communi- cations. Eliminating handover failure is very important. Some works are studied about handover issues[3-5].

For a handover, several steps are processed. In the handover measurement phase, MSs measure the signal strength of their serving cell and the neighboring cell. The

* Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 818 – 827, 2006.

measurement results are compared against the predefined thresholds and then it is decided whether to initiate the handover or not.

The possibility of whether MS is really going to out of the current cell or not can be decided by speed, direction, the time of connection duration and movement pattern of the MS. In the conventional researches about the handover procedure, only some of these elements have been considered. Basically, the signal strength coming from neighbor BSs are stronger than the predefined thresholds, the handover procedure is initiated. But, it can't be predicted whether the MS will really go out of the current cell or not. If the moving direction and speed of MSs are considered, it is possible to discriminate and ignore unnecessary channels for handover connections. Therefore, efficient direction and speed based handover connection control schemes are required to increase the utilization of channels and reduce useless efforts.

A straightforward and simple method for obtaining handover area is another key element to simplify handover rate analysis. In the previous research, a. Kim[6] proposed a simple method for obtaining handover area by concentric circles on cellular systems. Many approaches have been proposed for handover rate analysis[7,8]. Cho and Kim obtained handover rate using the speed of MSs, the size of the cell and the density of connections[8]. These studies did not consider the direction of MSs. In this paper, we propose an easy method for obtaining the handover rate using the speed and direction of MSs and our method focus on the soft handover.

The remainder of the paper is organized as follows. Section 2 presents the conventional handover rate using the size of the handover area. Section 3 proposes the handover rate using direction and speed-based MSs. Section 4 proposes numerical methods for obtaining the number of handover connections for both the conventional handover method and the proposed method and analysis of the results. The results from the simulation are presented in section 5. Concluding remarks are made in section 6.

2 Conventional Handover Rate

2.1 A Handover Process

An MS requests a soft handover to a neighboring BS when the pilot strength that received from the neighboring BS exceeds the handover threshold. If the handover request is accepted by the neighboring BS, the MS holds two channels assigned by the current BS and the neighboring BS. In fig. 1, when an MS detects that pilot strength exceeds T_ADD, it sends a handover request message to the neighboring BS and then it acquires a new traffic channel from neighboring BS. When the pilot strength drops below T_DROP, the traffic channel assigned by the current BS is released and eventually the MS can communicate with the only one[10].

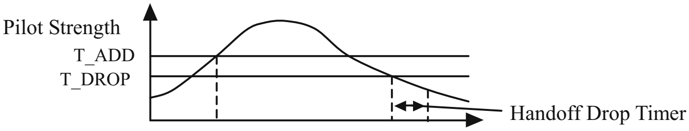


Fig. 1. Handover thresholds by pilot strength

2.2 A Cell and a Handover Area Model

An omnidirectional BS transmitter has a circular coverage area, defined by the area at which the received BS signal power exceeds a certain threshold. If the areas are circle and and all are of the same size, they can be assumed the concept of hexagonal cell. In actual practice, the coverage area by a particular BS may be not circular shape because the propagation loss is affected by natural and manmade terrain. But, we assume the model as a cell as the hexagon, because the idealized hexagonal cellular geometry provides us a simple method for analysis of cellular networks.

A cell boundary can be determined by the received BS signal power exceeding a certain threshold. In fig. 2, a cell is surrounded by six neighbors. We define two additional boundaries with the cell boundary to analyze the handover rate. The Hs(Handover Start) boundary is the point where the received pilot strength becomes over the T_ADD and the He(Handover End) boundary is the point where the received pilot strength becomes under the T_DROP.

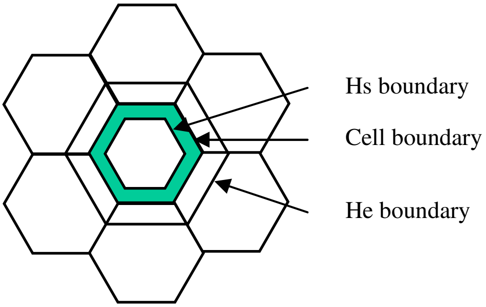


Fig. 2. Handover areas and cell boundaries

The handover area in a cell is the area from the Hs boundary to the cell boundary in fig.2. The dark area serves as a part of handover area for analysis of soft handover rate.

2.3 The Conventional Handover Rate by the Proposed Handover Area Calculation

The conventional handover channel rate can be easily obtained by the ratio of the size of handover area to the size of cell area. To get the size of handover area, we obtain a subset of a cell which is modeled as a hexagon which is the set of triangles within a hexagon.

The handover area in a cell is the area between the Hs boundary and the cell boundary. In fig.4, it is presented by the set of triangles within the area between the Hs boundary and the cell boundary. The size of the handover area can be obtained by multiplying the size of a triangle to the number of triangles within the handover area. The number of triangles for one side of handover area at the hexagon is 5 and the total number of triangles is 6×5=30. The side length of a triangle is a cell_radius/3 at fig. 4.

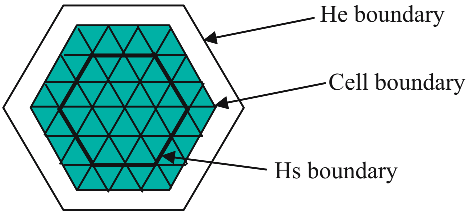


Fig. 3. An area modeled of a cell

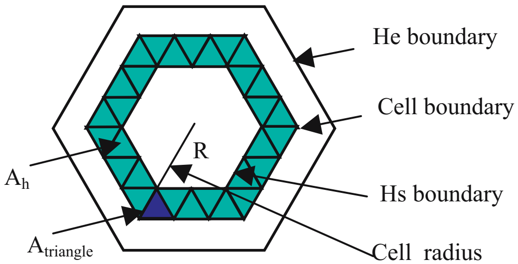


Fig. 4. A handover area model

As a general expression, when the side length of a triangle is R(cell_radius)/n, the number of triangles for one side of a handover area in a hexagon cell is 2n-1. Therefore, the total number of triangles in a handover area becomes 6× (2n-1).

Therefore, the size of a handover area A_h(dark area in fig. 4) is obtained by multiplying the number of triangles of handover area and the size of a triangle($A_{triangle}$).

$$\begin{aligned} A_h &= 6 \times (2n-1) \times A_{triangle} \\ &= 6 \times (2n-1) \times \frac{\sqrt{3}R^2}{4n^2} \cdot \end{aligned} \tag{1}$$

The obtained size of a triangle is $A_{triangle} = \frac{\sqrt{3}R^2}{4n^2}$.

In the basic model, we assume that a connection origination rate is uniformly distributed over the mobile service area.

We denote that the conventional handover rate R_h is obtained by using the ratio of the handover area to the cell area.

$$R_h = \frac{A_h}{A_c} \cdot$$

Since the size of a cell A_c is

$$A_c = \frac{3\sqrt{3}R^2}{2} \cdot$$

The conventional handover rate R_h is presented by

$$R_h = \frac{6\times(2n-1)\times \sqrt{3}R^2 / 4n^2}{3\sqrt{3}R^2 / 2} = \frac{2n-1}{n^2} \qquad (2)$$

3 Direction and Speed-Based Handover Rate

An MS requests a soft handover to a neighboring BS whenever the received pilot strength from the neighboring BS exceeds the T_ADD. But, the MS which requests a soft handover may not go out from the current cell. It is generally decided by the direction and speed of MS. If the moving direction and speed of MSs are considered, it can be predicted whether the MS will really go out of the current cell or not.

3.1 Direction and Speed Based Handover Rate

When a connection reaches the Hs boundary, the MS requests a soft handover. At this point, a handover probability of whether the MS will really go out of the current cell or not can be obtained. The direction and speed based handover rate is obtained under the following assumptions. In fig.5, 'D' is the distance that MS can move within the remaining time from the Hs boundary and is calculated by the speed of MS and theremaining time. 'v' is the speed of MS and t is remaining time which is the difference between the mean connection duration time and the elapsed time. The current BS can obtain the mean connection duration of the MS by averaging the connection duration time before.

In fig. 5, MS m_1 is originally occurred at inside of the Hs boundary and has zero probability to go beyond the He boundary, because it may not move more than the distance 'D_1' during the remained connection time. An m_2 can have some probability to go beyond the He boundary, if it moves within the angle θ. The probability of MS's going out the He boundary is θ/180. We assume that all MSs are moving straightly for outbound.

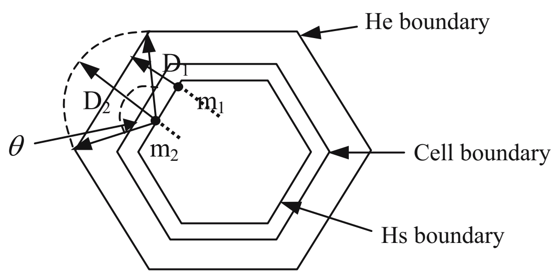


Fig. 5. An example of outgoing

For an easy calculation of the handover probability, we consider only one side of a hexagonal cell as fig. 6. when MS passes through the Hs boundary, it detects that pilot strength (E_c/I_t) exceeds T_ADD and requests a soft handover to a neighboring BS. At the same time, the possibility of MS going out of the He boundary can be obtained from 'θ'. It is presented by θ/180.

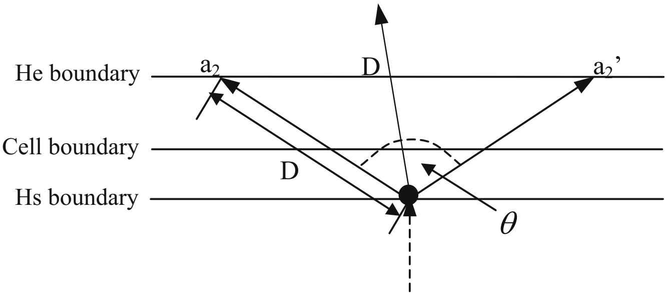


Fig. 6. Remodeling of an outgoing MS

To get the angle θ, we use the function of arc cos about a right triangle as fig. 7.

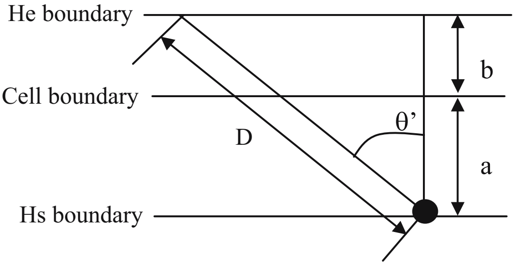


Fig. 7. A right triangle model

Where 'a' is the distance from the Hs boundary to the cell boundary and the 'b' is the distance between the cell boundary and the He boundary.

Since the θ' is $\cos^{-1}\dfrac{a+b}{D}$ in fig. 7, the outgoing handover probability at the Hs boundary is presented by

$$p = \frac{2\theta'}{180} = \frac{2\cos^{-1}\left(\dfrac{a+b}{D}\right)}{180}, \quad \textit{if} \;\; D > (a+b) \tag{3}$$

$$p = 0, \qquad\qquad \textit{otherwise.}$$

Since $\dfrac{2\cos^{-1}\left(\dfrac{a+b}{D}\right)}{180}$ is the outgoing handover probability of MS at the Hs boundary, the mean outgoing handover probability can be obtained by averaging the values of the handover probabilities at each positions $L_0, L_1, \cdots, L_n$ which are on the route of the MS in fig. 8. As 't'(remaining time) of MS at every position $L_0, L_1, \cdots, L_n$ is decreasing, the distance 'D'(v×t) is also decreasing.

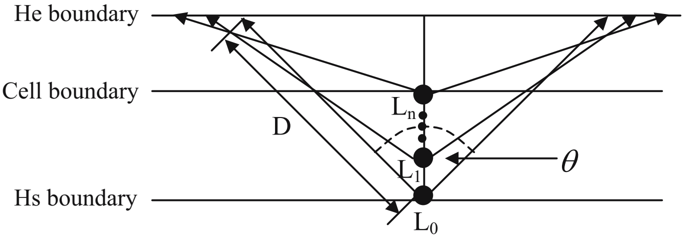


Fig. 8. An outgoing MS at positions $L_1, L_2, \cdots, L_n$

The mean outgoing handover probability is presented by

$$E(p) = \frac{1}{n+1}\sum_{i=0}^{n}\frac{2\cos^{-1}\left(\dfrac{b+a-\dfrac{a\times i}{n}}{D-\dfrac{D\times i}{K}}\right)}{180} \tag{4}$$

$$= \frac{1}{90(n+1)}\sum_{i=0}^{n}\cos^{-1}\left(\frac{b+a-\dfrac{a\times i}{n}}{D-\dfrac{D\times i}{K}}\right) \quad (n < K).$$

For all the connections that occurred in the current cell in fig. 9, the mean outgoing handover rate can be obtained.

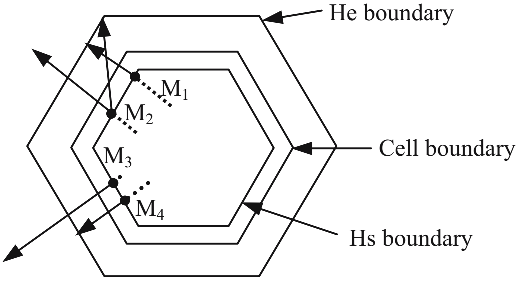


Fig. 9. An outgoing connection

The conventional handover rate is the number of existing connections in the area between the Hs boundary and the cell boundary. Therefore, the mean outgoing handover rate based direction and speed can be obtained by multiplying the mean outgoing handover probability and the conventional handover rate R_h.

The mean outgoing handover rate of a cell $E_{hr}(D)$ is presented as

$$E_{hr}(D) = \frac{R_h}{90(n+1)} \sum_{i=0}^{n} \cos^{-1}\left(\frac{b + a - \frac{a \times i}{n}}{D_i - \frac{D_i \times i}{K}} \right) \quad (n < K). \tag{5}$$

Where 'D_i' in (5) has a normal distribution with mean μ

4 The Number of Handover Connections

The number of conventional handover connections can be obtained by using the conventional handover probability R_h.

The mean number of conventional handover connections is

$$HN_C = R_h \times \text{the number of mean connections (connections/cell/s).} \tag{6 }$$

The mean number of outgoing handover connections based on direction and speed of MSs is

$$HN_{DS} = E_{hr}(D) \times \text{the number of mean connections (connections/cell/s).} \tag{7}$$

5 Experimental Results

We do computer simulations to verify the accuracy of calculation method of handover rate using GPSS/H. The assumptions for simulation are that a cell radius indicated by R=1km and parameters of the system are the same in any cell. The simulations are carried out under the following assumptions:

i) new connections are uniformly distributed in each cell area
ii) connection arrival follows Poisson distribution with mean arrival rate of λ
iii) a connection duration time has an exponential distribution with a mean of 60 second.

Table 1. The simulation result of conventional handover rate according to the change of a handover area

n	1	2	3	4	5	6	7	8	9	10
R_h	0.9999	0.7498	0.5519	0.4378	0.3587	0.303	0.2548	0.2323	0.21	0.189

The result shows the conventional handover rate R_h tends to decrease as handover area decreases. n is divisor of cell_radus(1 km).

Table 2. The numerical analysis of conventional handover rate according to the change of a handover area

n	1	2	3	4	5	6	7	8	9	10
R_h	1	0.75	0.56	0.44	0.36	0.31	0.27	0.23	0.21	0.19

Table 2 shows a numerical analysis using function(2).

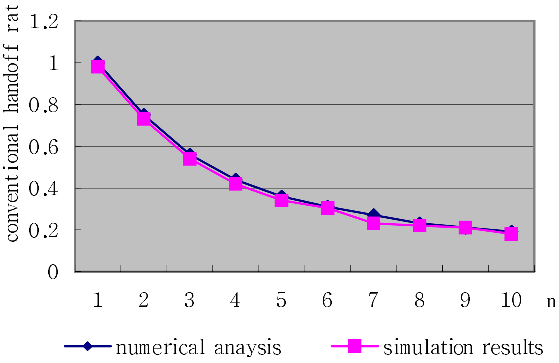


Fig. 10. A comparison of the simulation results and the numerical analysis of the conventional handover rate

The fig. 10 shows the comparison of the simulation results and the numerical analysis of the conventional handover rate(R_h) according to the size of a handover area. It shows that the result of the numerical model is in good agreement with the above simulation results.

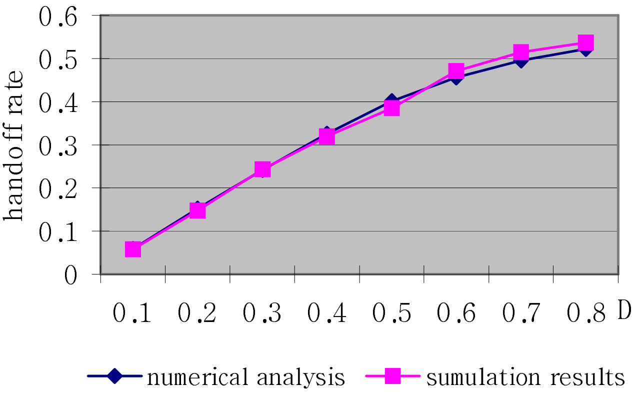


Fig. 11. A comparison of the numerical analysis and the simulation results

Fig. 11 shows a comparison of the simulation results and the numerical analysis of direction based handover rate.

We also have computer simulations to verify the accuracy of the proposed calculation method of direction based handover rate. It shows that the analytic model is in good agreement with the above simulation results.

6 Conclusions

We proposed a very simple and straightforward analytical handover model to estimate the soft handover rate based on direction and speed-based MSs. We model a cell as a hexagon which is the set of triangles and the handover area as the subset of triangles in the cell using the proposed calculation method. We can calculate the conventional handover rate using the handover area easily. We also propose the method of outgoing handover probability by using direction and speed-based MSs and the number of outgoing handover connections. Our analytic model is in good agreement with the computer simulation results. The proposed method for the handover rate may help to estimate the performance of a cell based mobile network system and can be easily applied to WiBro cellular networks.

References

1. Kim K., Ahn J. H., Kim K., "An Efficient Subcarrier and Power Allocation Algorithm for Dual-Service Provisioning in OFDMA Based WiBro Systems", International Conference, ICOIN 2005, Jeju Island, Korea, January 31- February 2, 2005, pp.725-734.
2. Kim Jae-Pyeong, Kim Do-Hyung, Kim, Sun-Ja, "Design of software architecture for mobile devices supporting interworking between CDMA and WiBro", the 7th International Conference on Advanced Communication Technology, 2005, ICACT 2005, v.1 pp.54-56.
3. Mahana Dhamayanthi Kulavaratharasah and A. H. Aghvami, "Teletraffic Performance Evaluation of Microcellular Personal Communication Networks (PCN's) with Prioritized Handover Procedures", IEEE transactions on vehicular technology, Vol. 48, No.1, Jan 1999.
4. S. S. Rappaport, "The multiple-call hand-off problem in high capacity cellular communication systems", Proc. IEEE Vehic. Technol. Conf., VTC '90, Orland, May 6-9, 1990, pp. 287-293.
5. S. S. Rappaport, "Models for call hand-off schemes in cellular communication networks", Winlab Workshop on Third Generation Wireless Information Networks, New Jersey, Oct. 18-19, 1990.
6. Chonggun Kim, Mary Wu, YungJun Choi, "A calculation method for soft handoff rate in cellular systems", 2000 Proceedings Ninth International Conference on Computer Communications and Networks, Las Vegas, USA, 2000, pp 653-656.
7. H.Xie and D.J. Goodman, "Mobility models and biased sampling problem", 1993 2nd IEEE International Conference on Universal Personal Communication Record, Vol.2, Oct 1993, pp. 803-807.
8. Moo-Ho Cho, Kwang-Sik Kim and Cheol-Hye Cho, "Anaysis of soft handoff rate in DS-CDMA cellular system", 1997 IEEE 6th International Conference on Universal Personal Communications Record, 1997, pp.235-238.
9. Daehyoung Hong and Stephen S. Rappaport, "Traffic Model and Performance Analysis for Cellular Mobile Radio Telephone Systems with Prioritized and Nonprioritized Handoff Procedure", IEEE transactions on vehicular technology, Vol. VT-35, NO.3, Aus 1986.
10. Vijay K. Grag, "IS-95 CDMA and cdma2000", pp.181-195, PRENTICE HALL.

The Classification of the Software Quality by Employing the Tolerence Class[*]

Wan-Kyoo Choi, Sung-Joo Lee, Il-Yong Chung, and Yong-Geun Bae[**]

School of Computer Engineering, Chosun University, Kwangju, 501-759, Korea
wankyoo@empal.com, ygbae@chosun.ac.kr

Abstract. When we decide the software quality on the basis of the software measurement, the transitive property which is a requirement for an equivalence relation is not always satisfied. Therefore, we propose a scheme for classifing the software quality that employs a tolerance relation instead of an equivalence relation. Given the experimental data set, the proposed scheme generates the tolerant classes for elements in the experiment data set, and generates the tolerant ranges for classfing the software quality by clustering the means of the tolerance classes. Through the experiment, we showed that the proposed scheme could product very useful and valid results. That is, it has no problems that we use as the criteria for classifing the software quality the tolerant ranges generated by the proposed scheme.

1 Introduction

Because the increase of the software cost of the total system cost, the interest in the software complexity and in the quality measurement has been augumented, and many measures have been proposed. The software measurement shows such inportant informations as maintanibility, understading and complexity of the software [1].

When we decide the software quality on the basis of the software measurement, we do it as the followings.

1. Define such linguistic variables as “it is easy to maintain” or “it is not easy to maintain”.
2. Decide the ranges of the measurement value corresponding to the linguistic variables.
3. Decide that any software belongs to a specific liguistic variable according to the measurement value.

When we decide that the structure of any program is more complex than is necessary because the cyclomatic number of it is more than 20, we apply the above process to our decision.

[*] This study was supported by research funds from Chosun University, 2004.

[**] Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 828–834, 2006.

Many researches [4, 7, 9, 11] suggest the various ranges for classifing the software quality, but they are based on an equivalence relation. When we decide the software quality based on the software measurement, the transitive property which is a requirement for an equivalence relations is not always satisfied.

For example, we define two liguistic variables about LOC(Line Of Codes), "complex" and "non-complex", and choose the ranges corresponding to two liguistic variables as "under 20" and "20 or more", respectively. In this case, we must decide that any program whose LOC is 19 is complex and the other program whose LOC is 20 is non-complex. However, it is more reasonable to decide that two programs have an similar degree of the complexity rather than the above decision.

This problem can happen whenever we decide the software quality by using an equivalence relation. This problem can be solved by using a tolerance relation [2, 3, 10] instead of an equivalence relation. Therefore, in this paper, we propose a scheme which generates the tolerant ranges for calssifing the software quality. We shows that we can generate the tolerant ranges for classifing the software by applying the tolerance relation to the representation of the similarity relation of the data.

2 Tolerance Relation

If data x, y and z satisfy the equivalence relation, they must satisfy the following properties.

1. The reflexive: ${}_xR_x$
2. The symmetric: ${}_xR_y \rightarrow_y R_x$
3. The transitive: ${}_xR_y\ and\ {}_yR_z \rightarrow_x R_z$

In case of classifing the data, the transitive property which is a requirement for an equivalence relations is not always satisfied. Thus it is not reasonable to represent the similarity of the data on the basis of an equivalence relation [2, 13]. For example, let x, y and z the elements of any data set. When x and y belong to the same linguistic variable and y and z belong to the same linguistic variable, x and z does not always belong to the same linguistic variable. This case always occurs on the boundary area, on which two linguistic variable are adjacent. Therefore, in case of classifing the data, we must represent the similarity relation between the data by a tolerance relation, which satisfy only the reflexive and the symmetric [3].

When we classify the software quality on the basis of the measurement value by the software measures, the same problem happens as classifing the data. This case also does not always satisfy the transitive. Therefore, the similarity relation between the software must be represented by the tolerance relation.

Let U be the universal set of the data, τ be the tolerance relation about any property, and $T(x)$ be a set of the elements having the tolerance relation with x. Then $T(x)$ is defined as the following [3, 13].

$$T(x) = \{y \in U \mid x \ \tau \ y\} \tag{1}$$

In general, a tolerance relation is represented by the similarity measure, which shows an degree of the similarity between two elements [5]. The similarity measure can is variously defined according to the addressed issues, but its general property is as following: Let the similarity measure between the property values of two data objects be $s(x,\ y)$. Then, when $s(x,\ y) > \alpha$ two data object x and y is said to have the tolerance relation. α is decided according to the addressed issues and is used to judge whether two data objects have the tolerance relation or not[13]. The tolerance class $\tau(x_i)$ of any element $x_i \in U$ is defined by using the similarity measure as like eq. 2.

$$\tau(x_i) = \{x_j \mid s(x_i, x_j) > \alpha,\ \ x_i, x_j \in U,\ j \neq i,\ j = 1, 2, \cdots, n\} \cup \{x_i\} \qquad (2)$$

3 Scheme for Generating the Tolerant Ranges

We propose a scheme for generating the tolerant ranges for classifing the software quality. Figure 1 shows our scheme. When the tolerent classes are given, it generates k tolerant ranges from them.

In step 1, we obtains the tolerent classes by using the similarity measure. The tolerent class for any data object x is the set of elements whose degree of the similarity to x is more than α. The number of the tolerant classes generated from n data objects is n.

In step 2, we calculates the means of the elements which belong to each tolerant class.

In step 3, we cluster the means from step 2 as k groups and obtain the tolerant ranges for classifying the software quality.

In step 3, we employ K-Means algorithm proposed by MacQueen. This algorithm classifies the data objects into k clusters, calculates the center value from the means of the data objects which is belonged to the cluster, calculates an distance from the center values to each data object, and includes each data objects into the cluster of the most near distance.

By the process of figure 1, we can get k ranges corresponding to k liguistic variables which satisfy the tolerance relation. When we classify programs by using the measurement value by the software measure, we can decide what tolerant

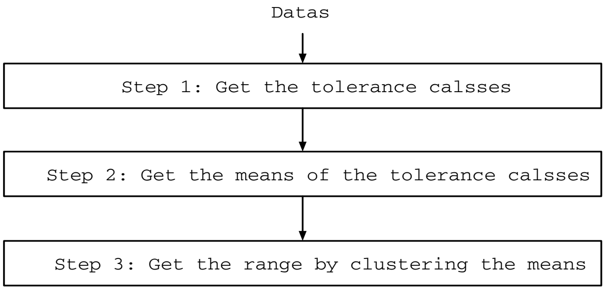

Datas
Step 1: Get the tolerance calsses
Step 2: Get the means of the tolerance calsses
Step 3: Get the range by clustering the means

Fig. 1. The process for generating the tolerant ranges

range they belongs to. That is, a program is classified into $G_i(i = 1, 2, \cdots, k)$, which is a linguistic variable for classifing the software.

4 Experiment

We applied our scheme to the set of the LOC measurement values about the modules written by C-language and generated the tolerant ranges for classifing the software quality by LOC. The modules were obtained from the source code of Linux kernal and Ansi-C runtime libray. The number of modules were 18404, and the total lines of all modules were 533165.

In general, the similarity measure is defined by using such distance functions as Hamming distance, Euclidian distance, etc.. However they cannot be applied when classifing the program objects on the basis of the measurement values by the software measures because they cannot reflct the phychological distance.

For example, when we calssify the programs by using LOC, any program of 10 lines and the other program of 20 lines have the obvious phychological difference, but any program of 100 lines and the other program of 110 lines does not have as obvious phychological difference as the former.

Therefore, we define the similarity measure for LOC by using the fuzzy membership function. Let U be a universal set for LOC values of the programs and x_{max} be the pre-determinated maximum value. The similarity measure between $x_i \in U$ and $x_j \in U$ is defined as like eq. 3.

$$\begin{aligned} s(x_i, x_j) &= \mu(x_i, x_j) \wedge \mu(x_j, x_i) \\ u(x_i, x_j) &= \frac{1}{1+(x_j-x_i)^2(x_{max}+1-x_i)/X_{max}}, \;\; j \neq i, \; j = 1, 2, \cdots, n \end{aligned} \tag{3}$$

When we assume $x_{max} = 100$, figure 2 shows an degree of the similarity between a program of 10 lines and the others and between a program of 90 lines and the others. In this figure, the range of the similar programs to a program of 10 lines and the range of the similar programs to a program of 90 lines are obviously different. Also, larger the value of LOC is, wider the range of the similarity is.

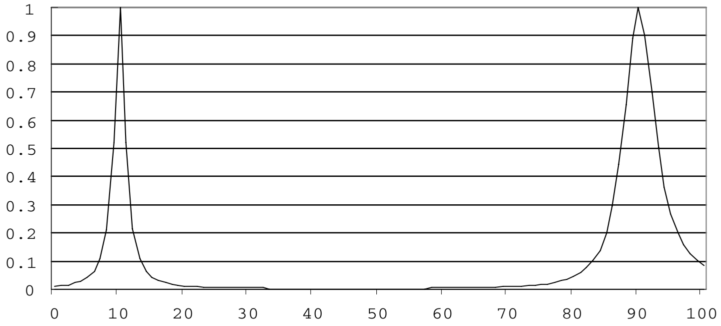


Fig. 2. An degree of the similarity between a program of 10 lines and the others and between a program of 90 lines and the others

When the size of a population is more that 20000 and the confidence interval is 95% and the tolerant error is within ±1%, the optimal sample size is 8213 [8]. Thus, we randomly retrieved 8213 modules from the experimental set consisted of 18404 modules, and made them the experimental group T_A, and made the rest the experimental group T_B. We generated the tolerant ranges for classifing the software from T_A, applied the generated ranges to T_A and T_B, and compare the result from T_A with the one from T_B.

The maximum value of LOC of the experimental set was 100. When α was 0.1 and the number of the linguistic variables were "small program", "medium program" and "large program", we could obtain the tolerant ranges corresponding to each linguistic variable as like table 1 from T_A.

Table 1. The obtained tolerant classification categories for LOC

Small program	Medium program	Large program
$0 <= \cdots <= 31$	$26 <= \cdots <= 61$	$54 <= \cdots <= 100$

Table 2. The number of programs in T_A and T_B

Experimetal group	Small program	Medium program	Large program
T_A	5426	2863	1089
T_B	6681	3413	1416

When the tolerant ranges of table 1 was applied to T_A and T_B, the number of modules belonging to each linguistic variable in T_A and T_B were as like table 2.

For testing the goodness of the tolerant ranges of table 1, we compared the characteristics of two experimental groups, T_A and T_B. That is, we compared the characteristics of the data set belonged to same linguistic variables by table 1. For comparing the characteristics, we tested that the classified data sets statistically had the significant difference by inferring the difference of the population mean of them.

When inferring the difference of the population mean, two assumptions are needed. The first is the assumption of a nomal distribution, that is, the distribution of the sample is approximately normal. The second is the assumption of the homogeneity of variances, because we can't previously know the poplation variance in most cases.

Table 3 shows the descriptive statistics of T_A and T_B, which are generated by SPSS. We can know that the distribution of the experimental groups is approximately normal on the basis of the central limit theorem and of the fact that the skewness and the kurtosis are close to 0 in table 3.

Table 3. Descriptive statistics of T_A and T_B

Experimetal group	Mean	Std. Deviation	Skewness	Kurtosis
T_A	29.1093	20.6437	1.269	1.092
T_B	29.1127	20.8118	1.269	1.035

Table 4. Test of Homogeneity of Variances

Categories	Levene statistic	df1	df2	Sinificance probalility(p)
Small program	2.248	1	12105	0.134
Medium program	4.507	1	6275	0.340
Large program	0.202	1	2503	0.653

Table 5. ANOVA Combined Between Groups

Categories	Sum of Squares	df	Mean Square	F	Significance probability(p)
Small program	35.607	1	35.607	0.719	0.397
Medium program	166.389	1	166.389	1.666	0.197
Large program	202.570	1	202.570	1.217	0.270

Let σ_A^2 and σ_A^2 be the population variances of T_A and T_B, respectively. We retrived Levene's statistic in order to test the assumption of the homogeneity of variances($H_0 : \sigma_A^2 = \sigma_A^2$). Table 4 shows Levene's statistic when the confidence interval is 95%. In each case, we cannot reject H_0 because Levene's statistics are enough large and p(=significance probability) $>$ 0.05(=significance level) [12].

Let μ_A^2 and μ_A^2 be the population means of T_A and T_B, respectively. By the result of table 4, we can know that the assumption of the homogeneity of variances is satisfied. Therefore, under the assumption of the homogeneity of variances, we performed the one-way ANOVA(ANalysis Of VAriance) in order to test that the classified data sets statistically had the significant difference($H_0 : \mu_A^2 = \mu_A^2$). Table 4 shows ANOVA statistics when the confidence interval is 95%. In each case, we cannot reject H_0 because p(=significance probability) $>$ 0.05(=significance level) [12].

Therefore, we could conclude that the characteristics of two experimental groups is not different when calssifying two experimental groups by table 1. This shows that our scheme for generating the tolerant ranges can product very useful and valid results only if the given assumptions can be satisfied and that the ranges retrieved by it can be used as the criteria for calssifying the software quality.

5 Conclusion

When we classify the software on the basis of the measurement values by the software measures, the transitive property which is a requirement for an equivalence relation is not always satisfied. Therefore, in this paper, we propose a scheme for calssifying the software quality by the tolerance relation which satisfies the reflexisive and the symmetric.

Our scheme obtains the tolerent classes by using the similarity measure, and calculates the means of the elements which belong to each tolerant class, and cluster the means as k groups, and obtains the tolerant ranges for classifying the software quality.

In experiment, we applied our scheme to LOC values of 18302 modules written by C-language. We obtained the tolerant ranges from an experimental set, applied the obtained ranges to two experimental groups, and compared their characteristics. As result, we could conclude that their characteristics is not different, that is, the obtained ranges can be used as the criteria for calssifying the software.

References

1. Karl J. Ottensteion, Linda M. Ottensteion, "The program dependence graph in a software development environment", ACM SIGPLAN Notices, vol. 19, no. 5, pp. 177–184, May 1984.
2. A. Patel, "Transformation functions for trapezoidal membership functions", Internat. J. Comput. Cognition, vol. 2, pp. 115–135, 2004.
3. K. Funakoshi and T.B. Ho, "Information retrieval by rough tolerance relation", The 4th international Workshop on rough sets, Fuzzy sets, and Machine Discovery, Tokyo, Nov. 1996.
4. Horst Zuse, Software Complexity-Measures and Methods, New York: Walter de Gruyter, pp. 25–37, 1991.
5. M. Kretowski and J. Stepniuk, "Selection of objects and attributes a tolerance rough set approach", ICS Research Reports, 1994. H. Pan and C.H. Yeh, A metaheuristic approach to fuzzy project scheduling. In: V. Palade, R.J.
6. Howlett and L.C. Jain, Editors, Knowledge-based Intelligent Information and Engineering Systems, Springer, pp. 1081–1087, 2003
7. Lowell J. Arthur, Measuring Programmer Productivity and Software Quality, New York: John Wiley & Sons, pp. 138–142, 1985.
8. D.J. Luck, H.G. Wales, D.H. Taylor, Marketing Research, N.J.: Prentice-Hall, pp. 611–612, 1970.
9. T. McCabe, "A Complexity Measure", IEEE Trans.SE., SE-2, pp. 308–320, 1976.
10. Slowwinski R. and Vanderpooten D. "Similarity relations as a basic for rough approximations, ICS Research Reports, 1994.
11. Szentes J., Gras J., "Some Practical Views of Software - Complexity metrics and a Universal Measurement Tool", First Australian Software Engineering Conference, Canberra, pp. 14–16, May 1986.
12. John Neter, William Wasserman and Michael H. Kutner, Applied Linear Statistical Models, IRWIN, Boston, 1990.
13. Daijin Kim and Chul-Hyun Kim, "Handwritten Numerical Character Recognition Using the Tolerant Rough Set", Journal of Fuzzy logic and Intelligent Sytems, vol. 9, no. 1, pp. 113–123, 1999.

Components Searching System Using Component Identifiers and Request Specifics

Jea-Youn Park[1], Gui-Jung Kim[2], and Young-Jae Song[1]

[1] Dept. of Computer Engineering, Kyunghee University,
1, Sochen-dong, Gihung-gu, Yongin-si,
Gyeonggi-do 446-701, Republic of Korea
jy_bak@khu.ac.kr, yjsong@khu.ac.kr

[2] Department of Bio Medical Engineering, Konyang University,
26, Nae-Dong, Nonsan, Chungnam, Republic of Korea
gjkim@konyang.ac.kr

Abstract. This paper defines agreement relation between use-cases of UML and component request specifics by classifying and specifying non-standardized shaped specifics of each procedure step in order to search appropriate component that operates required role from user in the request analyzing step. Using the agreement relation defined in this paper between request specifics and component specifics can automatically search appropriate component by calling request specifics file without user's direct inquiry information. To improve trade off between Recall and Precision has abstracted by using component glossary which has a Lexical Chain addition. A plan has been presented that decrease development term and cost and also maximizes reusability, the strength of component system, by applying candidate component which was extracted at the beginning of system development.

1 Introduction

Established studies [1][2] had some problems lowered flexibility and efficiency because they search component which is corresponded with signature by extracting class information attributes such as class names, class member functions, class attributes after the steps of analysis and design. This paper has presented how to search appropriate component which executes required roles from the user in the step of component identification before in order to solve these problems. Therefore, it is able to increase efficiency and cut down developing cost. The research method to search component efficiently by extracting component identifiers are as followed.

- To search component wanted in the step of requests analysis specifies use-cases of UML and each procedure step by classifying like <object>, <medium> <function>, <postInforamation>, <preIn-formation>, etc.
- It defines compliance relations of each to search component automatically. It will be done by classifying use-case items of request specifics, procedure items, component items of component specifics, and method items into multi facets.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 835–843, 2006.

- It is presented as multi facets by presenting summarized specifics and detailed specifics through syntax analysis and semantic analysis assembling information by using functional and non-functional information, and aspect of component.
- It divides component searching procedure into step1 and step2. And then it defines 1^{st} similarity to execute 1^{st} search through summarized specifics of component and 2^{nd} similarity for 2^{nd} search in order to improve precision by using detailed specifics. When it searches, it uses component glossary that is added lexical chain including relation information such as synonyms, parent words, and child words.

Specifics about requests and components can manage data effectively with providing structural shapes by standardizing in the shape of XML based on natural languages. And it can search components automatically by reading request specifics files without user's direct input. In this paper, in order to improve trade off between Recall and Precision analyzed syntax and semantic. And it abstracted these analyses by using component glossary which added lexical chain with the notion of synonyms, parent words, and child words. It also presented a method that could maximize reusability, an advantage of component system, and reduce a term of development and cost.

2 Related Work

In this chapter, we would like to describe a method that measures similarity component to established component specification method and standardized specific.

2.1 A Method of Component Specification

Component specification helps to identify solve these problems by defining what component should provide and expect, and also provides a method that improves reusability and makes an easy approach. Also, by defining component specification, it could build an early architecture through these, established architectures are considered.

C2 Architecture. C2 [3], a component developed from UCI, is a kind of message based architectures which are for software system with flexibility and extension. It has an architecture like fig1. C2 styled component is divided into request messages which call a method declared inside by receiving messages via top port and bottom port, and notification messages which generates outgoing messages. It is impossible to communicate directly between components in C2 styled component. It interacts by exchanging messages between top port and bottom port of component and between top port and bottom port of connect. ADL which supports the technology of C2 styled architecture designed and specified and ADN and IDN which have similar syntax form to JAVA.

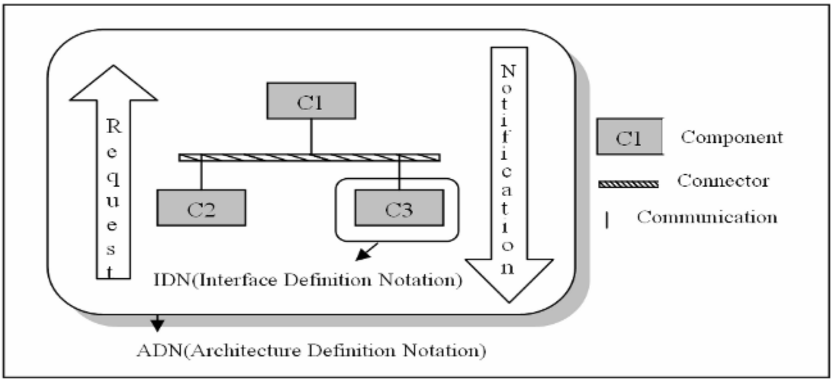

Request
C1
C2
C3
Notification
IDN(Interface Definition Notation)
ADN(Architecture Definition Notation)
C1 Component
Connector
Communication

Fig. 1. C2 Architecture

Component Specification Method by Classifying Component Service by Aspect. In order to inquiry component on the base of abstracted classified features specified by classifying the service which component provides and requires to by various component aspect. Component has aspect detail and aspect detail property [4][5].

- Aspect detail.
  To specify service information about User interface, Distribution, collaborative work, security, transaction processing, and persistency management.
- Aspect detail property.
  To specify non-functional information such as User interface element, transaction throughput, performance of distributed system, and restriction particular.

Fig 2 shows aspect detail which is described in Distribution aspect the service that Event Transport Component provides and requires. '−'is a service required from component and '+' is a service provided from component. Standardized specification computes similarity between components by using predicate. In [6], predicate expresses precondition, postcondition, performance and function of component and helps user's understanding of component. Elements of predicate are CLASS, DATA MEMBER, and MEMBER FUNCTION. Attributes of elements are as followed.

- CLASS (name[,inherited mode, inherited class name])
- DATA_MEMBER (name, #data type, #access mode)
- MEMBER_FUNCTION (name, #return type, #access mode
  #void | [,parameter name, #data type]):VIRTUAL(#0 | #1)

Similarity formula is as followed by using predicate consisted as shown above.

$$s = \frac{\sum e_{ij}}{\sqrt{n\sqrt{m}}}$$

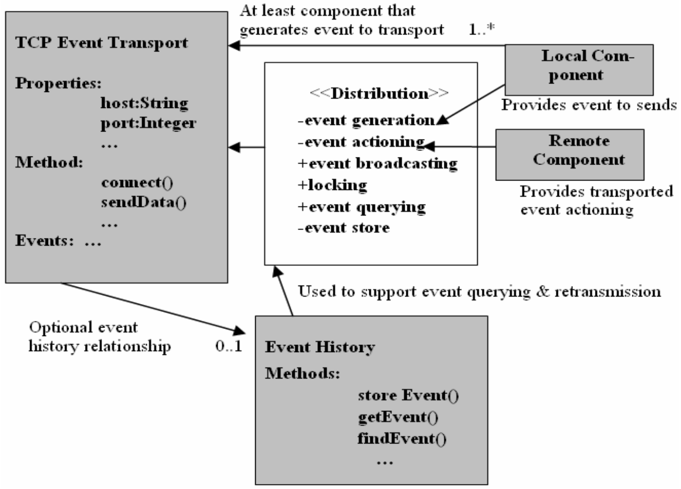


Fig. 2. Event Transport component aspect detail

$e_{ij} = 1$ if $a_i = b_j$ $e_i j = 0$ if $a_i \neq b_j$
for i = 1....n and j =1....m
a_i : predicate of component A b_j : predicate of component B

3 Component Identifier Generation and Request Specification by Using Lexical Chain

In this paper, intermediate form of specification has been used to standardize by abstracting summarized specification and detailed specification of component in order to make developer easy to specify non-specification based on natural language and to convert and integrate data. First, it classifies and divides component specification into non-functional information, functional information and assembly information. And abstracting component identifier from component specification is to use for search.

3.1 Component Specification

Component specification items are largely described <generalInfo> which shows non-functional information, <functionInfo> which shows functional information and <assemblyInfo> which shows assembly information by using aspect.

3.2 Generation of Component Identifier

Lexical chain is a presentation of strata relations of synonyms, parent words and child words by organizing into notions or meanings. Features of component from meaning analysis are consisted of facets by specifying serial and combined

Table 1. Component Specification Items

Component Architecture	General Info	Platform	
		Size	
		ProgramLanguage	
		price	
	Function Info	Component	ComponentName
			ComponentSpec
			ComponentFunc
			ComponentObj
			ComponenMed
		Method	MethodName
			MethodSpec
			MethodFunc
			MethodObj
			MethodMed
			MethodPara
			MethodRe
	Assembly Info	UserInerface	UserInerfaceRequire
			UserInerfaceProvide
		Distribution	DistributionRequire
			DistributionProvide
		Collaborative	CollaborativeRequire
			CollaborativeProvide
		Security	SecurityRequire
			SecurityProvide

meaning information of each vocabulary. To do so, it selects basic vocabulary used basically and uses lexical chain.

Component identifier constitutes gathering basic classes that are classified facet unit in order to present component which it wants to classify. The architecture of component identifier presented in this paper has classified non-functional and functional information as summarized specification and detailed one, specified not only classifying code but also assembly information as aspect in order to present lexical chain. And also it has presented component identifier as a form of multi facets by using these classifying attributes.

– Component Classifier
  <platform, size, programLanguage, componentName, ...,>
– Method Classifier
  < methodName, methodFunc, methodFuncCode, methodObj, methodObjCode, methodMed, methodMedCode, ..., >
– Aspect Classifier
  < userInterfaceRequire, userInterfaceProvide, distributionRequire, distributionProvide,collaborativeWorkRequire, ..., >

Table 2. Requests Specification Items

Usecase Architecture	Special Require	UsecasePlatform	
		UsecaseSize	
		UsecaseLanguage	
		Usecaseprice	
	Usecase Function	Usecase	UsecaseName
			UsecaseSpec
			UsecaseFunc
			UsecaseObj
			UsecaseMed
		Process	ProcessName
			ProcessSpec
			ProcessFunc
			ProcessObj
			ProcessMed
			ProcessPreInformation
			ProcessPostInformation

3.3 Requests Specification for Inquiry Information Extraction

For accurate analysis among requests, we used a use-case diagram of UML and then defined by classifying each item of diagram through the step of vocabulary analysis like as it shown table 2.

4 Components Searching System and Similarity

4.1 Components Searching System

Fig3 shows the procedure of components searching based similarity of agreement between requests specification and component specification.

4.2 Similarity

To improve trade off between recall and precision, we classified 1st search for component summarized search and 2nd search for component method search. The improved formula is as shown below in order to get similarity with soon-to-be-generated candidate component.

Def 1 Q_1 : 1^{st} user query
qi : summarized specification attributes generated
through usecase modeling

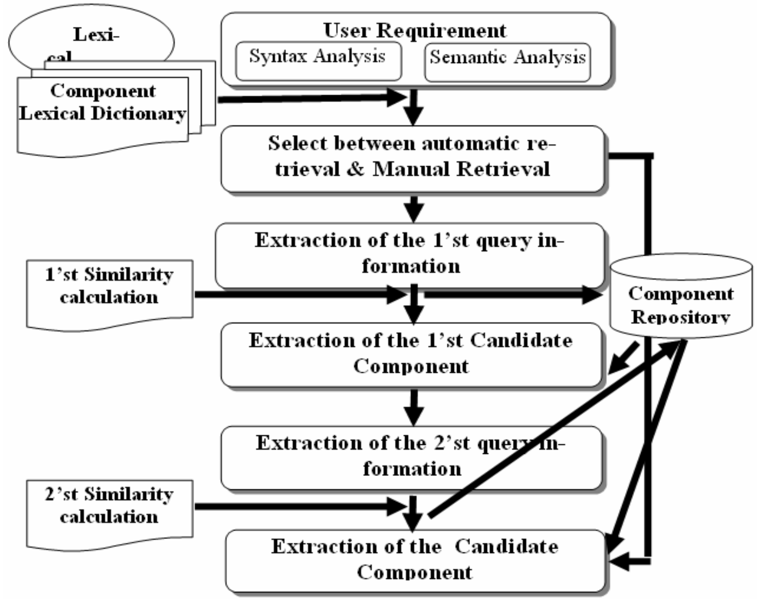


Fig. 3. Components Repository Construction and Searching Procedure

(usecasePlatform, usecaseSize, usecaseLanguage, use-casePrice, usecase-Name, usecaseFunc, ...)
$Q_1 = (q_1, q_2, q_3......., q_l)$
Def 2 A : component
a_i : component classifier
(platform, size, programLanguage, price, componentName, compoFunc, compoObj, compoMed)
$A = (a_1, a_2, a_3.........., a_l)$

1^{st} similarity)

$$P_1 = \frac{\sum e_i}{l}$$

$e_i = 1$ if $a_i = q_j$
$e_i = 0$ if $a_i \neq q_j$
for i = 1....l

In order to raise precision of component search, 2nd query should be executed by using method classifier between usecase process attribute and component identifier. The formula that gets a similarity between user's query and extracted candidate component is as followed.

Def 3 Q_2 : 2^{nd} user query
q_i :detailed specification attribute generated through usecase procedure modeling
(processFunc, processObj, processMed, processPreIn-format..)
$Q_2 = (q_{l+1}, q_{l+2}, q_{l+3} \ldots\ldots, q_m)$

Def 4 B : extracted candidate component from the result of 1^{st} query
b_i : method classifier of extracted candidate from the result of 1^{st} query (methodFunc, methodObj, methodMed, methodPara, methodRe, methodName, methodFunc, ...,)
$B = (b_{l+1}, b_{l+2}, b_{l+3} \ldots\ldots\ldots, b_n)$

2^{nd} similarity)

$$P_2 = \frac{\sum e_i}{l}[\frac{\sum S_{jk}}{(m-l)(n-l)}]$$

$S_{jk} = 1$ if $b_j = q_k$
$S_{jk} = 0$ if $b_j \neq q_k$
for j = l+1...n, k = l+1...m

5 Conclusion

In this study, we have designed a system that manages data efficiently with providing an architectural form and searches components automatically without user's direct input by reading request specification file. For doing so, it should standardize appropriate request specification and component specification in order to search the component that performs the role required by the user in the procedure of request analysis which is just before component identifying procedure into the form of XML based of natural language. Through the searching system which is consisted of 1st and 2nd procedure, similarity will be defined. And by using it, the most similar component candidates will be shown in order. Also, to improve trade off between Precision and Recall, it makes limits-control of query terms possible by using component glossary which added lexical chain of synonyms, parent/child words through syntax and semantic analysis. This study is only able to show the search result as a form of text. It makes understanding of whole components low. Therefore, in the future, graphic view like component diagram should be needed, and standard component glossary should be required to improve the precision of search.

References

1. R.Prieto-Diaz, "Implementing Faceted Classification For Software Reuse" Comm. ACM, vol. 34, no. 5, pp. 89–97, May, 1991
2. Kim, Haengkon and Cha Jungeun, "A Study on the component understanding system for supporting of the object-oriented prototyping", The Transactions of Korea Information Processing Society, Vol. 4, No. 6, 1997, 1519–1530.

3. P. Oriezy, N. Medvidovic and R. N. Tayleor, “Architecture Definition Language,” Proceed-ings of the International Conference Software Engineering, 1999.
4. John Grundy, “Storage and retrieval of software components using aspects”, Computer Science Conference, 2000. ACSC 2000. 23rd Australasian , 2000
5. Houssam Fakih, Noury-Bouraquadi, International Workshop on Aspect-Oriented Software Development
6. Chao-Tsun Chang, Chu, W.C., Chung-Shyan Liu, Hongji Yang, “A Formal Approach to Software Components Classification and Retrieval” Computer Software and Applications Conference, COMPSAC ’97 proceeding, 1997.

Software Architecture Generation on UML

Haeng-Kon Kim

Department of Computer Information & Communication Engineering,
Catholic University of Daegu, Korea
hangkon@cu.ac.kr

Abstract. The on going underlying work aims to provide a robust and straight forward basis to the UML for modeling and analysis. In the context of architecture driven software development approaches, UML has become the most useful specification language for the systems. In this paper, we are concerned about the SAGU(Software Architecture Generation on UML) methodology to guide the reflexive development of architectures from the software requirements. In particular, we are detailing the first step of this methodology and the definition of the goals model whose constituents are the fundamental basis for the overall process defined in SAGU proving its suitability for obtaining traceable architectural models.

1 Introduction

In spite of the fact that the OMG [1,2] has adapted the UML for productivity and quality for systems, every new development new issues appear related to the customization of the software. They attempt to accomplish either environmental or the stakeholders' needs that are evolving over time. How to overcome deficiencies and limits exhibited by traditional development methodologies is a clear challenge that has been faced by several techniques. Aspect Oriented Software Development (AOSD) [3,4] is related to the first type of techniques providing advantages in expressiveness by the separation of concerns. It has many advantages related to software qualities. Both functional and non-functional needs, such as performance or compatibility, of the system's behavior can be separately acquired and specified across the development lifecycle.

Concerning the second category of techniques, existing approaches provide high-level architectural descriptions that are well suited for runtime evolution. Architectural reflection [5] is a clear example in this sense, offering facilities for dynamic reconfiguration, in terms of composition and/or connection, to existing systems without modifying them. Several approaches [6] make use of a meta-level in such a way that software systems are aware of their architectural properties and are able to adapt themselves at runtime in a simple way.

In both cases, however, evolving stakeholder's requirements give rise to the need of adaptability. It is inherent to every software development that expectations about the system are evolving over time. The Requirements Engineering process (RE) establishes the foundation on which the system-to-be should be implemented.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 844 – 852, 2006.

Therefore, it has to be able to identify and define this need of adaptability into its artifacts in such a way that they can be traced to low level abstraction artifacts.

A methodology SAGU (Software Architecture Generation on UML) [5,6] illustrates a process to concurrently define requirements and architectural artifacts. In this paper, we are detailing the first step of this methodology, i.e., the definition of the goals model whose constituents are the fundamental basis for the overall process defined in SAGU.

2 Related Works

2.1 Component Architecture

Component architecture is required to formula interactions among components and defines standardizations of common interface. So, it is necessary to layout guidelines for production, delivery, acquisition and assembly of componentsNow, various architectures are focusing on location of components and composition among them.

We have suggested **ABCD**(**A**rchitecture Platform/**B**ase Application Component/**C**ommon Business Component/**D**omain Component) architecture as standard model for evolving CBD process[7]. It refers to existing distributed computing reference model such as Sanfrancisco and CORBA reference model and is layered by scope, abstraction and granularity for integrating of multi solutions. Layer A and B are responsible to API for middles and basic formats for distributed object services. So, layer B contains existing distributed objects such as CORBA, EJB. Layer C supports common functionalities across multi domains with divided into "Common part" and "Core part". Also, it supplies assembling resources and includes customizable patterns for constructing a business framework. Layer D includes specific application components categorized by vertical sub-domain within special area. (Figure 1) represents overall structure of ABCD architecture.

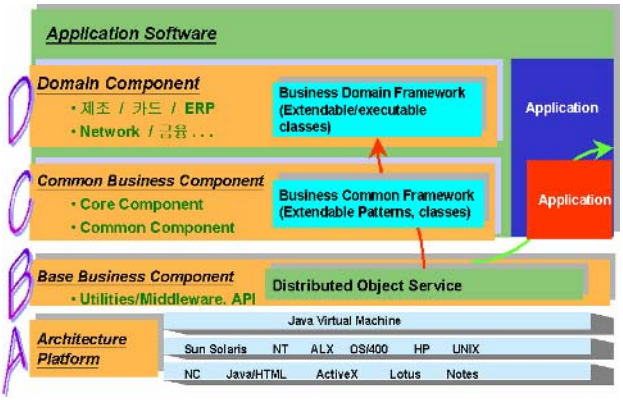

Application Software
D
Domain Component
• 제조 / 카드 / ERP
• Network / 금융 . . .
Business Domain Framework (Extendable/executable classes)
Application
C
Common Business Component
• Core Component
• Common Component
Business Common Framework (Extendable Patterns, classes)
Application
B
Base Business Component
• Utilities/Middleware. API
Distributed Object Service
A
Architecture Platform
Java Virtual Machine
Sun Solaris NT ALX OS/400 HP UNIX
NC Java/HTML ActiveX Lotus Notes

Fig. 1. ABCD Software Component Architecture

2.2 Current States in CBD

CBD has become rapidly substantial interesting field in the business application. Specially, CBD is primarily used as a way to assist in controlling the complexity and risks of large-scale system development, providing an architecture-centric and reuse-centric approach at the build and deployment phases of development. So now, many vendors and researchers have tried to establish the CBD maturity by involving followed strategies: 1) efficient building of individual components, 2) efficient building development solutions in new domain, 3) efficient adapting the existing solutions to new problems and efficient evaluating of set of solutions. But, by the lack of standardization and clearness for CBD approach method, we can't expect a practice benefits in business solution.

3 Software Architectures on UML

3.1 SAGU Phases

This methodology is concerned with the definition of software architectures from functional and non-functional requirements. With this aim, it provides the analyst with a guidance along the process from an initial set of requirements to an architectural instance. SAGU is a not a product oriented approach but a process oriented one. It does not describe a set of metrics to determine if the requirements, functional and non-functional, are implemented into the final software. On the contrary, requirements and architectural design decisions are defined concurrently in such a way that every decision is make to achieve a given requirement.

SAGU iterates over a set of five steps (Figure 2) which are described next.

Phase 1: Goals Model (GM) Definition

In this step, the set of goals to be accomplished by the software system are defined by means of both functional and non-functional requirements. An informal set of requirements, stated in natural language, is the input to trigger the model definition. The elaboration of the Goals and Scenarios Models are two intertwined processes. The GM is operationalized at the next step by means of the Scenarios Model (SM). Furthermore, the analysis information provide by the SM helps us to refine and identify new goals at the next iteration.

Phase 2: Scenarios Model (SM) Definition

The analyst has to identify the set of scenarios which operationalize the established goals and compose them to form an iterating and branching model of the system's behavior. Each scenario depicts the elements that interact to satisfy a specific goal and their level of responsibility in achieving a given task. These elements are *shallow-components*, i.e., a rough description of the components that appear into the final software architecture.

Use Cases and Message Sequence Charts are employed for the construction of the SM. Several alternative scenarios can operationalize the same goal, just like several alternative programs can implement the same specification. In order to offer the best approach, they have to be analyzed caring about conflicts that may arise among the operationalized goals.

Phase 3: Collaborations Definition
The third step is aware of the collaborations among the identified shallow components. The collaborations realize to UCs through collaboration diagrams.

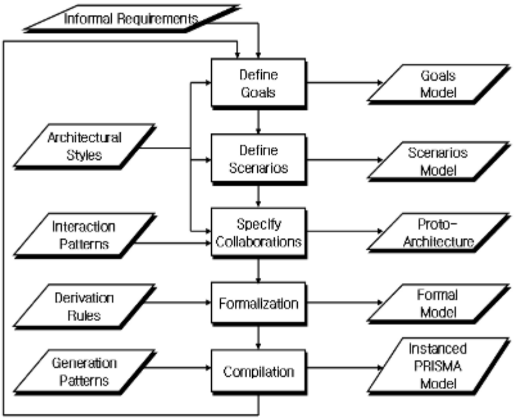


Fig. 2. Phase for Software Architecture Generation on UML

Additionally, the connectors are defined according to the components interactions. These first class citizens are required to achieve a loose coupling between the components.

Phase 4: Formalization
A semantic check and analysis of the models and the proto-architecture is required to identify eventual conflicts, e.g. different scenarios resulting in incompatible architectural configurations, and obtain the best alternative to avoid (or minimize) them. A set of derivation rules are provided to generate a formal specification from scenarios, goal and collaborations, in order to assist and speed up the formalization process. This specification is validated and then used for an automatic compilation process in the next step.

Phase 5: Compilation
Using the formal model and a set of generation patterns, the translation from the requirements model to an instantiated. This architectural model is able to be compiled into a concrete target system preserving its compiled into a concrete target system preserving its reflexive properties. This step is assisted by the engineer in order to refine the architectural elements. As SAGU is intended to be iterative and incremental, a feedback is provided from Step 5 to 1. In this way, all the models are up-to-date all over the process.

3.2 Generation Architecture Using Goal Model

Architectural models are a bridge between requirements and the system-to-be providing us with a lower abstraction level. They are used as intermediate artifacts to analyze whether the requirements are met or not. Therefore, this paradigm has two advantages that make it appropriate to systematically guide the selection among several architectural design alternatives:

- Its ability to specify and manage positive and negative interactions among goals allows the analyst to reason about design alternatives.
- Its capability to trace low-level details back to high-level concerns is very appropriate to bridge the gap between architectural models and requirements.

These are the main reasons why the *Goals Model* has been introduced as an SAGU artifact to identify and describe the users' needs and expectations, their relationships and how these can be met for the target system.

3.3 Components for the Goals Model

As was stated above the GM provides a number of abstractions in terms of which constraints on the software systems have to be defined. A key element employed in its construction is a goal.

- *Functional goals* describe services that the system provides, i.e., the transformations the system performs on the inputs.
- *Non-functional goals* refer to how the system will do these transformations, for instance, in term of performance, adaptation, security, etc. We are highlighting them because they are especially meaningful in terms of software quality.

Additionally, other aspects have to be stated when a goal is defined. For instance, a set of *preconditions* and *postconditions* has to be identified, Preconditions establish which situations must hold before some operation is performed. Postconditions define the situations that have to be achieved after some operation. Their evaluations help us to determine the best design alternative among those that satisfy the postconditions for the established goals.

Moreover, each goal has to be classified according to its priority, from *very high* to *very low*, for the system-to-be. This classification helps the analyst to focus on the important issues. These priorities can arise from several factors: organizational ones when they are critical to the success of the development, constraints on the development resources.

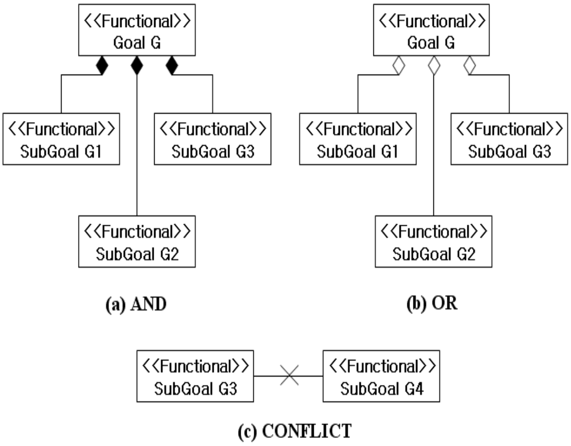
```
GOAL GoalName
ID identifier
TYPE [functional | Nonfunctional]
DESCRIPTION ShortDescription
PRIORITY [Very High | High | Normal | Low, Very Low]
AUTHOR autherName
PRE conditions
POST conditions
```

They must offer a set of solutions that allow the system to achieve the established goals. These solutions provide architectural design choices for the target system which meet the user's needs and expectations. They are called operationalizations because they describe the operation of the system, i.e., the system behaviour, to meet functional and non-functional requirements.

OPERATIONALIZATION *Opname*
ID identifier
DESCRIPTION ShortDescription
AUTHOR authorName
PRIORITY [Very High | High | Normal | Low, Very Low]

3.4 Relationships: An Element in the Refinement Process

There are two types of refinements that can be applied: intentional and operational. The former describes how a goal can be reduced into a set of subgoals via AND/OR relationships. The latter depicts how a set of solutions address a goal by means of AND POERATIONALIZE/OR OPERATIONALIZE relationship. Both, building blocks and relationships are structured as an acyclic goal graph, where the refinement is achieved along the structure, from the higher to the lover level, by applying intentional and operational refinements as in figure 3.

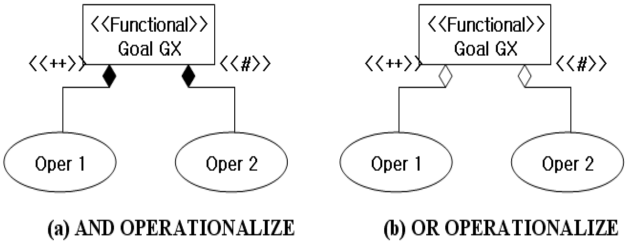

<<Functional>> Goal G
<<Functional>> SubGoal G1
<<Functional>> SubGoal G3
<<Functional>> SubGoal G2
<<Functional>> Goal G
<<Functional>> SubGoal G1
<<Functional>> SubGoal G3
<<Functional>> SubGoal G2
<<Functional>> SubGoal G3
<<Functional>> SubGoal G4

(a) AND

(b) OR

(c) CONFLICT

Fig. 3. Visual notation for Goals Relationships



(a) AND OPERATIONALIZE

(b) OR OPERATIONALIZE

Fig. 4. Visual notation for Operationalize Relationships

Every goal, which is too coarse-grained to be directly addressed by a solution, is refined in a set of subgoals which are a decomposition of the original one. Whenever a sub-goal is needed to achieve a goal, and AND relationship is established between them. On the contrary, if the sub-goal may optionally appear, then an OR relationship is established between them.

4 Case Study

Several Functional and Non-Functional Goals can be established from these sentences. They provide us an oversimplified view of this kind of applications, but which will allow us to understand how GM is defined and its relationship with the concerns of the system-to-be. Specially, it will show how GM and SM are interleaved along the process.

Iteration 1. The initial set of Requirements is presented as five nodes in Figure 5 obtained from the previous sentences. The *Performance* requirement is due to high efficiency required from this kind of applications. *Adaptation* appears because of the self-configurability or auto-adaptability that is needed. Finally, *Security* is introduced to deal with information protection.

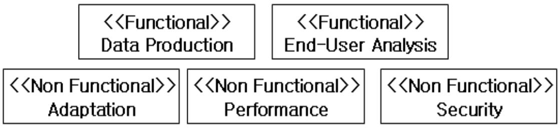


Fig. 5. Original Quality Goal Model

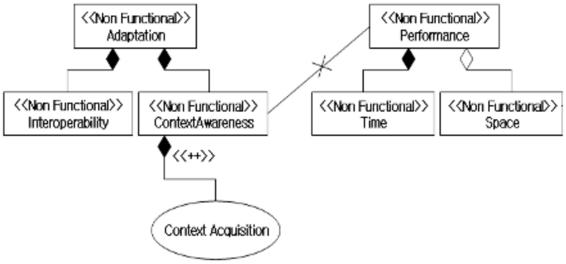


Fig. 6. Iterated Goals Model (part of)

Iteration 2. Several subgoals can be identified in order to satisfy the previous ones, as it is shown in Figure 6. For instance the *Adaptation* goal can be intentionally refined up to *ContextAwareness* and *Interoperability.*

Iteration 3. *– Step 1. Goals Model Definition.* Some of the identified goals can be operationally refined. *ContextAwareness* and *ContextAcquistion* are AND OPERATIONALIZE-related because the latter is a required solution for the former (Figure 7).

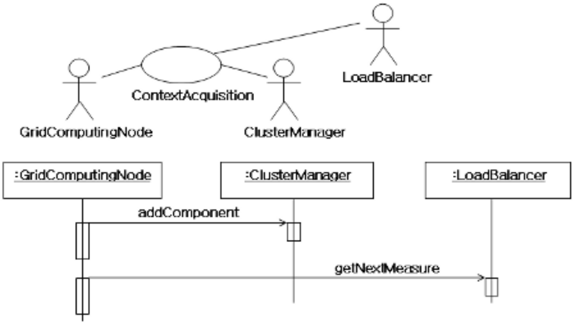


Fig. 7. Iterated Scenarios Model (part of)

Iteration 3. *– Step 2. Scenarios Model Definition.* In order to operationalize the subgoal *ContextAware,* three actors can be identified: *GridComputingNode, Cluster Manager* and *LoadBalancer*. The first one has to gather context information and distribute it to the LoadBalancer and the ClusterManager in order to optimize the resources. UC and the MSC, for this sub-goal are showed in Figure 8.

Iteration 3. *– Step 3. Collaborations Definition.* Because MSCs are closely related to collaboration diagrams it is very easy to get the latter. The result for the ContextAware goal can be observed in Figure 9.

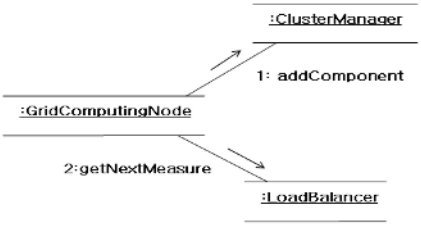


Fig. 8. A view of collaboration

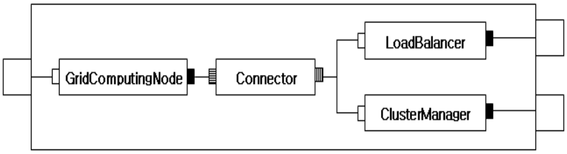


Fig. 9. A preview of the Instantiated

Iteration 3. *– Step 4. Formalization.* The defined models will be formalized in this step, but a detailed description is outside of the scope of this paper.

Iteration 3. *– Step 5. Compilation.* According to the information provided by the previous models and the proto-architecture.

5 Conclusions and Future Work

Summarizing, in this paper we show how to address the iterative development of requirements and architectures during the development of software systems. A methodology, SAGU, that guides the analyst, from an initial set of requirements to an instantiated, architecture has been presented. It uses the strength provided by the coupling of scenarios and goals to systematically guide through the iterative process. Moreover, it allows the traceability among both artifacts to avoid lacks of consistency.

Additionally, requirements and architectures can evolve iteratively and concurrently, in such a way that running-systems can dynamically adapt their composition and/or topology to meet their evolving requirements.

Acknowledgement

This research work has been supported by the IT Research Center (ITRC) Project in 2005.

References

1. http://www.omg.org/
2. J. Suzuki and Y. Yamamoto, "Extending UML with Aspects: Aspect Support in the Design Phase", AOP Workshop at ECOOP'99, Lisbon, Portugal, 1999.
3. AOSD Davy suvee," JasCo: Aspect-Oriented Approach Tailored for Component Based Software Development," AOSD conference , pp. 21–29, 2003
4. http://www.aosd.net
5. J. Perez, I. Ramos, J. Jaen and P. Letelier, E. Navarro: "PPRISMA: Towards Quality, Aspect Oriented and Dynamic Software Architectures", 3rd IEEE International Conference on Quality Software, Dallas, Texas, USA, November 6–7, 2003.
6. A. I. Anton, "Goal-Based Requirements Analysis", Proc. 2nd Int. Conf. on RE, Colorado Springs, CO April 15–18, 1996.
7. Haeng-Kon Kim, Jung-Eun Cha, Ji-Young Kim, Eun-Ju Park, Identification of Design Patterns and Components for Network Management System_, SNPD '00 International Conference, Vol. 1, No. 1, pp. 426–431, May, 2000
8. E. Navarro, I. Ramos and J. Perez: "Requirements and Architecture: a marriage for Quality Assurance". 8 Jornadas de Ingenieria del Soft. Y Bases de Datos. November 12–14, 2003.
9. B. Nuseibeh, "Weaving the Software Development Process Between Requirements and Architecture", Proc. 1st Int. Workshop From Software Requirements to Architectures (collocated ICSE), 12–19 May 2001, Toronto, Ontario, Canada.

Distributed Programming Developing Tool Based on Distributed Object Group Framework

Chang-Won Jeong[1], Dong-Seok Kim[2], Geon-Yeob Lee[3], and Su-Chong Joo[2]

[1] Research Center for Advanced LBS Technology of Chonbuk National University, Korea
mediblue@chonbuk.ac.kr
[2] School of Electrical, Electronic and Information Engineering,
Wonkwang University, Korea
{loveacs, scjoo}@wonkwang.ac.kr
[3] Dept. of Automotive & Mechanical Engineering, Kungang College, Korea
gylee@kunjang.ac.kr

Abstract. We are to suggest the GUIs of Distributed Programming Developing Tool(DPD-Tool) based on the Distributed Object Group Framework (DOGF). The 3 GUIs we implemented are the user interfaces between the DOGF and distributed program developers. In this paper, we explain first of all the DOGF and distributed programming tool we developed before. And then, for convenient development of distributed applications, we design and implemented 3 GUI environments such as the object group administrator, server program developers, and client program developers, by interactions among 3 GUIs. Finally using above environments, we showed the procedures for developing distributed applications and the result of execution of a distributed application implemented as an example under 3 GUI environments supported by DPD-Tool.

1 Introduction

With the advent of high-speed networking distributed environments, the distributed systems have been using for sharing various distributed resources and for providing wide-area critical-mission services. For this reason, distributed system environments are required the complicated interactions for improving availability of distributed resources and for obtaining promptly response from the other systems[1,2]. For supporting above issues, we need to solve the complicated interactions through effective management of distributed resources, like object group management, and provide simple binding and real-time strategies for increasing the availability of resources located on distributed systems[3]. The solutions of these requirements have been researched on area of the distributed middleware and platform. Especially, many researchers are interested in the group management of distributed objects for supporting a logical single view system environment and reducing interactions among them. For achieving and satisfying these interesting researches, we developed the Distributed Object Group Framework(DOGF) [4,5,6]. And after this, we also developed the Distributed Programming Developing Tool(DPD-Tool) based on DOGF. This Tool-Kit can be supported not only any middleware and any programming language, and object group management but also functionalities of the DOGF. The researching area

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 853 – 863, 2006.

in this paper, the GUI Environment of Distributed Programming Developing Tool(DPD-Tool), is shown in Figure 1.

According to given figure above, we define functions and interactions among each component of the DOGF and the DPD-Tool. For supporting convenient user interfaces, while we are developing distributed applications by using the DPD-Tool. We designed and implemented the 3 GUI environments for the object group administrator, server program developers, or/and client program developers and interactions among 3 GUIs. Finally using above environments, we showed the procedures for developing distributed applications using 3 GUIs, and the result of its execution in distributed systems given.

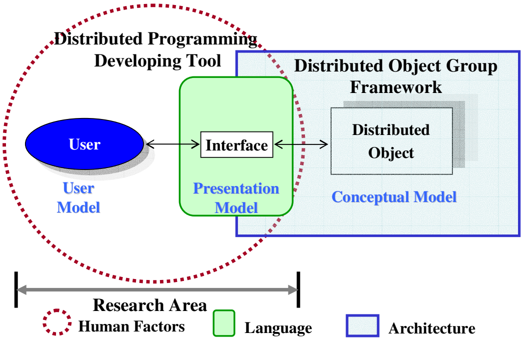


Fig. 1. The researching area in this paper

2 Our Previous Works

We have been studying the object-oriented technologies managing of the object group that can improve the executabilities of the distributed application by providing distributed transparency to clients. The DOGF is implemented between Communication & distributed middleware tier and distributed application tier. This framework supports 2 kinds of service; object group management service and real-time service. In our previous paper [7, 10], we have ever verified the executability of the supporting services by applying the DOGF to the distributed defense system as an example of distributed applications.

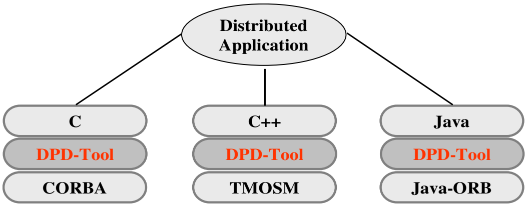


Fig. 2. Developing environments of distributed applications Using DPD-Tool

The DPD-Tool is supported by functionalities of the Distributed Object Group Framework(DOGF) via Application Program Interfaces(APIs) so that program developers can conveniently implement distributed applications. Figure 2 is showing the several programming environments of distributed applications using The DPD-Tool. This tool provides the developing environment of distributed application independently from any kind of application programming language and distributed middleware. The components supporting the object group management and the distributed real-time service in our tool are implemented to packaging DLL (Dynamically Linked Library).

The Group Manager object in the DOGF is responsible for wholly managing distributed objects including inside of an object group as the executing unit of distributed application, and is a unique object in an object group interfacing with client object. The distributed objects configuring an application take the group register/withdraw, the security check, the dynamic binding service and so forth via the functional interfaces implemented in the Group Manager object.

```
#include "DOGST_DLL.h"
class GroupManagerObject {
//register an object to the object group.
 char* enter_objectgroup(char *group_name, char
     *service_name, char *object_name, char
     *location_address);
//withdraw an object from the object group.
 char* withdraw_objectgroup(char *group_name, char
     *service_name, char *object_name);
//modify an object's info. registered in the object group.
 char* modify_objectgroup(char *service_name, char
     *group_name, char *object_name, char
     *location_address);
//insert access right of objects for client.
 char* insert_access_right(char *client_name, char
     *group_name, char *service_name);
//delete access right of client for objects.
 char* delete_access right(char *client_name, char
     *group_name, char *service_name);
//request the object's reference for executing service.
 char* request_object_infoToIRO(char *client_name, char
     *group_name, char *service_name);
};
```

Fig. 3. Functional interfaces of the Group Manager object in DOGF

Figure 3 shows the functional interfaces of the Group Manager object for managing group from this tool. The remaining components of the DOGF supporting in the DPD-Tool are implemented in the same way as the implementation of the Group Manager object. The detailed functions and the structure of components are referred to [4,5,6].

3 Development of GUIs of DPD-Tool

The distributed applications can be conveniently developed via 3 GUIs implemented in the DTD-Tool. While programming by using this tool, server or client program developers can develop the distributed application. For this, the DTD-Tool supports 3 Graphical User Interface (GUI) environments. The GUI for the object group administrator manages the total developing environment of distributed application. The second GUI using by server program developers is responsible for the group register/withdraw and the access right of objects of server program. And, finally the third GUI using by server program developers supports the developing environment of the client program that requests distributed service. By using these GUIs, the distributed application developers can conveniently use the supporting functions provided from the DOGF. Server program developers make a server program on specialized server systems, and register these service objects to the DOGF via GUIs of our tool. Figure 4 is showing the GUIs and interactions among distributed application, DPD-Tool and DOGF for implementing distributed applications.

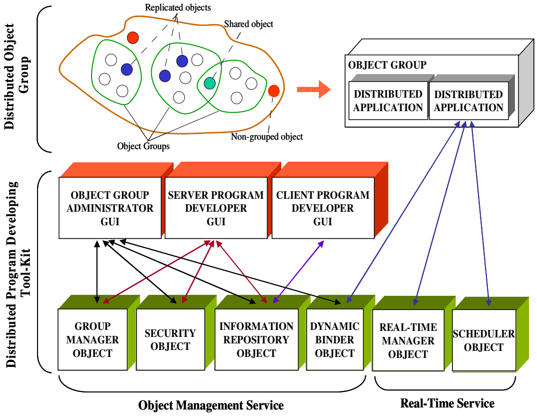


Fig. 4. GUIs and interactions among distributed application, DPD-Tool and DOGF

4 Development of Distributed Application Using GUIs of DPD-Tool

4.1 A Sample of Distributed Application

This section suggests and develops a distributed application with 4 operations(*add( ), subtract( ), multiple( ), divide( )*) using 3 GUIs environments supported by DPD-Tool based in the DOGF. These applications used the Time-triggered Message-triggered Object(TMO) scheme and TMO Supporting Middlware(TMOSM) [8,9]. These operations executing in a distributed applications are implemented to the TMO objects on 2 systems as follows; Add_TMO1, Add_TMO3, Subtract_TMO, and Multiple_TMO objects are located on System A, and Add_TMO2 and Divide_TMO objects are located on System B. Add_TMO1, Add_TMO2, and Add_TMO3 are replicated objects with the same service property, *add( )*, as we shown in Figure 5 that shows the sample of distributed application and its execution processes on distributed systems.

In the first step, ① the client requests the reference of service object executing the *add()* service to the Group Manager object. Next, ② the Group Manager object checks the access right of the object that provides the corresponding service for the client request. And, ③ if possible to access them, the Group Manager object requests the object's reference to the Information Repository object. The Information Repository object examines the replication status of objects. At this time, if the object exists non-replicated, the Information Repository object returns the object's reference immediately. ④ If the objects are replicated objects(Add_TMO1, Add_TMO2, and Add_TMO3), the Information Repository object requests the selection of object's reference for object binding to the Dynamic Binder object. After this, ⑤ the Group Manager object returns the object's reference(Add_TMO2) selected to the client. Finally, ⑥ the client requests the service to the object by referring to the object reference received from the DPD-Tool, and receives the executing results.

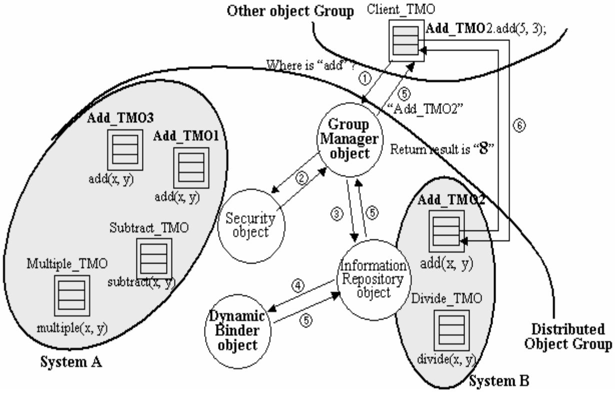


Fig. 5. Sample of distributed application and its execution processes

According to the clients' request, Binding algorithm must select an optimal object being on distributed systems with the minimum overhead from replicated objects. Before implementing of an application, these objects are managed by object group administrator, as a unit of group, *Operator*. Client program developers ought to take access right of these objects necessary to application from object group administrator or server program developer.

4.2 Developing Procedures of Distributed Application

In this section, we explain the developing procedure of a distributed application using GUIs environments supported by DPD-Tool. Firstly, the server program developers can implement server programs which given their systems using the server program developer GUI. After then, they register them into the Information Repository via the object group administrator GUI. Client program developer checks service objects with the access rights using client program developer GUI from the Information Repository stored service objects. In this time, when a client has not the access rights about the service object he/she needs, a client can request the access rights to the object group administrator newly. Finally, a client implements the client program using the permitted service objects in the DPD-Tool environment. In the next phase, when the program implemented by a client is executed using the service objects with their references obtained via invoking and returning properties of the permitted service objects. The Figure 6 below shows the developing procedures of a distributed application under the 3 GUI environments in the DPD-Tool.

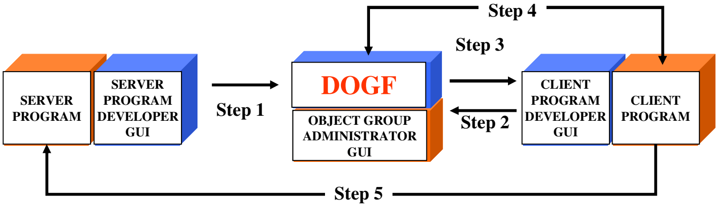
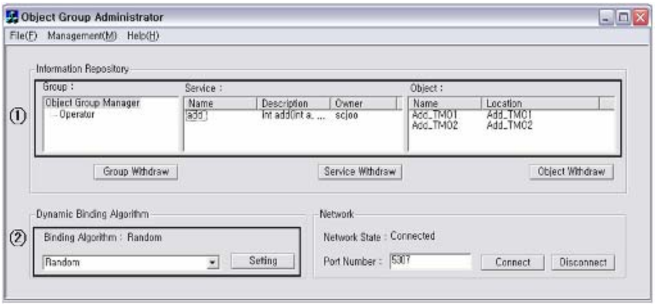

SERVER PROGRAM
SERVER PROGRAM DEVELOPER GUI
Step 1
DOGF
OBJECT GROUP ADMINISTRATOR GUI
Step 4
Step 3
Step 2
CLIENT PROGRAM DEVELOPER GUI
CLIENT PROGRAM
Step 5

Fig. 6. Developing Procedures of Distributed Application Using GUIs

4.2.1 GUI for the Object Group Administrator

This GUI supports the object group management supporting the DOGF of distributed objects to help the execution of distributed application. Figure 7 shows the GUI for object group administrator. In ①, we define the Information Repository for Groups, Services, and the implemented objects and their locations for services. These Groups, Services and Objects insert to the Information Repository by server program developers, they withdraw from the Information Repository by the object group administrator using each private GUI, respectively.



Information Repository

Group :

Object Group Manager
- Operator

Service :

Name	Description	Owner
add()	Int add(int a, ...	scjoo

Object :

Name	Location
Add_TM01	Add_TM01
Add_TM02	Add_TM02

Dynamic Binding Algorithm

Binding Algorithm : Random

Random

Network

Network State : Connected

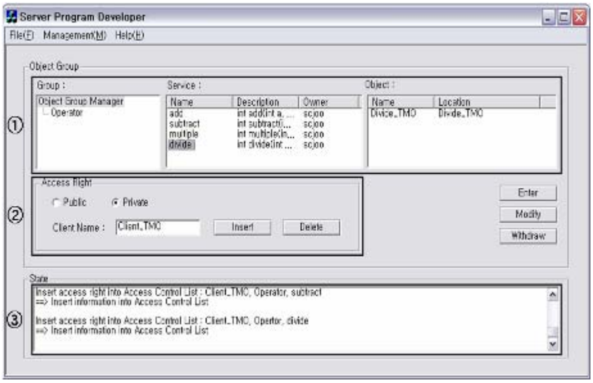
Port Number : 5307

Fig. 7. GUI for object group administrator

In details, And, ② is showing that we register/withdraw the objects into the given Group *Operator* using the service name, also grant the access right of objects to clients. In ②, we can use selectively binding algorithms for choosing one of the replicated objects on a group via supporting the DOGF.

4.2.2 GUI for Server Program Developers

The GUI for the server program developers is responsible for granting the access rights to clients and for implementing the components to execute distributed application, as service objects. In Figure 8, the Object Groups(*Operator*) Service(*add()*), and Objects(*add_TMO1, Add_TMO,* these objects are the same service property) are implemented by server program developers are showing in ①. In ②, the server program developers register objects implemented by him/her-self into the Group by the service name, also set the access right to the arbitrary clients for the corresponding services. The configuration and status information about the Object Group are displayed in ③.



Name	Description	Owner
add	int add(int a, ...	scjoo
subtract	int subtract(i...	scjoo
multiple	int multiple(in...	scjoo
divide	int divide(int ...	scjoo

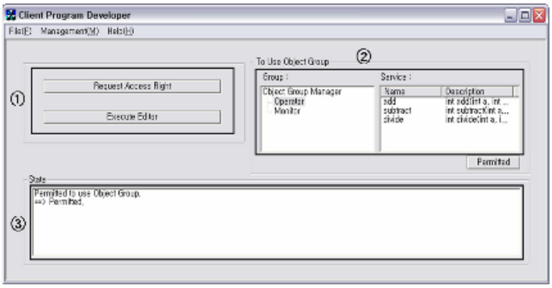
Name	Location
Divide_TMO	Divide_TMO

```
Insert access right Into Access Control List : Client_TMO, Operator, subtract
==> Insert information into Access Control List

Insert access right into Access Control List : Client_TMO, Opertor, divide
==> Insert information into Access Control List
```

Fig. 8. GUI for server program developer

4.2.3 GUI for Client Program Developers

In GUI for client program developer, it provides the distributed objects with the access right permitted to clients for developing distributed applications. By using ① in Figure 9, when client program developers want to use service object needed for developing a distributed application, they can request the access rights of services objects from server program developer or the object group administrator.



Name	Description
add	int add(int a, int ...
subtract	int subtract(int a,...
divide	int divide(int a, i...

```
Permitted to use Object Group.
==> Permitted.
```

Fig. 9. GUI for client program developer

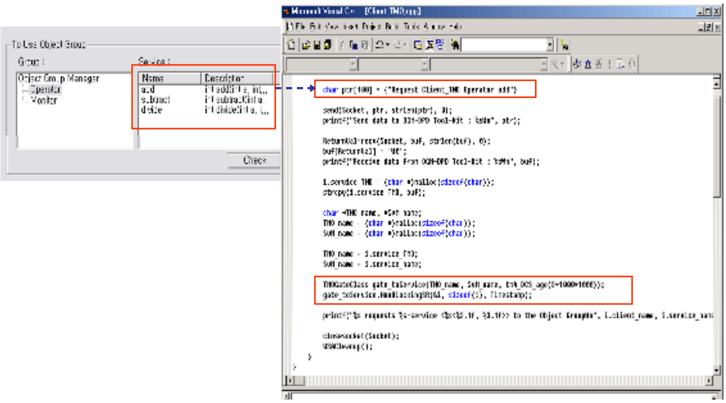
The button of "Execute Editor" calls an appropriate editor for writing client programs. The ② part is showing that client program developers are requesting the Groups(*Operator, Monitor*) and Services(*add(), subtract(0, divide()*) for developing client programs to server program developers or the object group administrator with pushing button of "Request Access Right". Finally, the ③ part is showing the Group permitted from server program developers or the object group administrator. In the next phase, a client can write a distributed application using the permitted service objects in Group on the appropriate programming editor, as we shown in Figure 7.

4.3 Development of Server Program Using GUIs of DPD-Tool

Each server program developer implements the service object based on TMO scheme providing the corresponding service on the server system using the server program developer GUI. And after then, they register them into the service object repository via the object group administrator GUI.

4.4 Development of Client Program Using GUIs of DPD-Tool

Via referring objects in a group shown on the client developer's GUI in Figure 10, the client program developers can make a distributed application independently without considering server environment.


To Use Object Group
Group :
Object Group Manager
Operator
Monitor
Service :
Check
Microsoft Visual C++ - [Client TMO.cpp]
File Edit View Insert Project Build Tools Window Help

Name	Description
add	int add(int a, int,..
subtract	int subtract(int a,..
divide	int divide(int a, i...

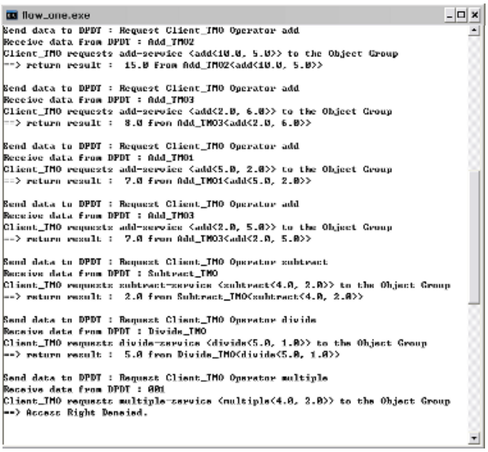
```
char ptr[100] = {"Request Client_TMO Operator add"}

send(Socket, ptr, strlen(ptr), 0);
printf("Send data to OGM-DPD Tool-Kit : %s\n", ptr);

ReturnVal=recv(Socket, buf, strlen(buf), 0);
buf[ReturnVal] = '\0';
printf("Receive data From OGM-DPD Tool-Kit : %s\n", buf);

i.service_TMO = (char *)malloc(sizeof(char));
strcpy(i.service_TMO, buf);

char *TMO_name, *SvM_name;
TMO_name = (char *)malloc(sizeof(char));
SvM_name = (char *)malloc(sizeof(char));

TMO_name = i.service_TMO;
SvM_name = i.service_name;

TMOGateClass gate_toService(TMO_name, SvM_name, tm_DCS_age(3*1000*1000));
gate_toService.NonBlockingSR(&i, sizeof(i), Timestamp);

printf("%s requests %s-service <%s(%3.1f, %3.1f)> to the Object Group\n", i.client_name, i.service_nam

closesocket(Socket);
WSACleanup();
    }
}
```

Fig. 10. TMO-based client program developed by C++

For this, the server program developers have to register the referred objects to the Information Repository object that is a component implemented in the DOGF. After then objects in the rectangle box of left side must to be declared in client program, like the type declaration of general programming language. In details, the *Operator* group and the *Print* group are registered in the DOGF. And the former group has the objects for executing *add( )*, *subtract( ), multiple( )*, and *divide( )* services and the latter group has the objects for executing *printer( )* and *monitor( )* services. The remaining parts for developing a client program are equal to the existing program mechanisms. Figure 10 is showing the examples of client programs implemented by the TMO-based C++.

4.5 Executing Results of Distributed Application

Figure 11 shows the executing results of the distributed application, i.e. client/server program, using service objects invoking by a client program. The client program requests the objects for executing *add( ), subtract( ),* and *divide( )* services in the *Operator* group from server systems, repeatedly. Here we assume that requesting client cannot take the access right for *multiple( )* service from a server program developer. In this figure, we showed the service results and their interactions while invoking objects in the distributed environment as shown in Figure 6. Considering that a client requests one of the replicated objects(Add_TMO1, Add_TMO2, and Add_TMO3) for receiving the result of *add( )* service, the DPD-Tool returns the appropriate object's reference out of replicated objects to the client by operating the dynamic binding service and the client requests the service by referring the returned object's reference.



```
Send data to DPDT : Request Client_TMO Operator add
Receive data from DPDT : Add_TMO2
Client_TMO requests add-service <add<10.0, 5.0>> to the Object Group
==> return result :  15.0 From Add_TMO2<add<10.0, 5.0>>

Send data to DPDT : Request Client_TMO Operator add
Receive data from DPDT : Add_TMO3
Client_TMO requests add-service <add<2.0, 6.0>> to the Object Group
==> return result :  8.0 from Add_TMO3<add<2.0, 6.0>>

Send data to DPDT : Request Client_TMO Operator add
Receive data from DPDT : Add_TMO1
Client_TMO requests add-service <add<5.0, 2.0>> to the Object Group
==> return result :  7.0 from Add_TMO1<add<5.0, 2.0>>

Send data to DPDT : Request Client_TMO Operator add
Receive data from DPDT : Add_TMO3
Client_TMO requests add-service <add<2.0, 5.0>> to the Object Group
==> return result :  7.0 from Add_TMO3<add<2.0, 5.0>>

Send data to DPDT : Request Client_TMO Operator subtract
Receive data from DPDT : Subtract_TMO
Client_TMO requests subtract-service <subtract<4.0, 2.0>> to the Object Group
==> return result :  2.0 from Subtract_TMO<subtract<4.0, 2.0>>

Send data to DPDT : Request Client_TMO Operator divide
Receive data from DPDT : Divide_TMO
Client_TMO requests divide-service <divide<5.0, 1.0>> to the Object Group
==> return result :  5.0 from Divide_TMO<divide<5.0, 1.0>>

Send data to DPDT : Request Client_TMO Operator multiple
Receive data from DPDT : 001
Client_TMO requests multiple-service <multiple<4.0, 2.0>> to the Object Group
==> Access Right Denied.
```

Fig. 11. Execution results of distributed application

Finally, the client receives the executing results from the object. We adapted the Random algorithm out of several binding algorithms to the Dynamic Binder object in the DOGF for selecting one object out of 3 replicated objects described above. From Figure 12, the first 4 requests remotely call the *add( )* service and receive each executing result of its service, as a sequence of Add_TMO2, Add_TMO3, Add_TMO1, and Add_TMO3 selected by Random algorithm. The 5th and final 6th requests invoke Subtract_TMO and Divide_TMO and receive each executing result of these services, respectively. Finally, the 7th request is denied due to not taking the access right from a server system about the *multiple( )* service.

5 Conclusions and Future Works

In this paper, we developed the 3 GUI environments for developing distributed application conveniently. These 3 GUIs provide the development environment for server/client program developers and the management environment for the object group administrator. These GUI environments contained in Distributed Programming

Developing Tool(DPD-Tool) based on the Distributed Object Group Framework (DOGF). Using these GUIs, we showed the developing procedure of distributed application and implemented client program and server programs on the DPD-Tool.

Through the experimental results of the distributed application developed on this DPD-Tool, we showed that it is possible to execute the group management of objects configuring the distributed application and the dynamic binding for the replicated objects, and to provide conveniently the easy developing environment for programing distributed applications via supporting this tool with 3 GUI environments. In future works, we are planning to add and extend the medical information service components and interfaces in the DPD-Tool based on a Healthcare Home Service Framework. Via extending functionalities of the DOGF, we are researching the Healthcare Home Service Framework integrating with healthcare devices, healthcare appliance, biosensors, and medical information systems on the ubiquitous computing environments.

Acknowledgements. The authors wish to acknowledge helpful discussions of the TMO Programming Scheme with Professor Kane Kim in DREAM Lab., University of California at Irvine. This research was supported by the Regional Research Centers Program of the Korean Ministry of Education and Human Resources Development through Center for Healthcare Technology and Development.

References

1. M. Takemoto: Fault-Tolerant Object on Network-wide Distributed Object-Oriented Systems for Future Telecommunications Applications. In IEEE PRFTS (1997) 139-146
2. P.M. Melliar-Smith, L.E. Moser, and P. Narasimhan: Consistent object replication in the Eternal System. Theory and Practice of Object System, Vol.4, No.2 (1998) 81-92
3. V. Kalogeraki, P.M. Melliar-Smith, and L.E. Moser: Dynamic Scheduling for Soft Real-Time Distributed Object Systems. In Proceedings of the IEEE 3rd International Symposium on Object-Oriented Real-Time Distributed Computing (2000) 114-121
4. C.S. Shin, M.H. Kim, Y.S. Jeong, S.K. Han, and S.C. Joo: Construction of CORBA Based Object Group Platform for Distributed Real-Time Services. In Proceedings of the 7th IEEE International Workshop on Object-oriented Real-time Dependable Systems(WORDS'02) (2002) 229-302
5. S.C. Joo, C.S. Shin, C.W. Jeong, and S.K. Oh: CORBA Based Real-Time Object-Group Platform in Distributed Computing Environments. Lecture Notes in Computer Science, Vol.2659 (2003) 401-411
6. C.S. Shin, M.H. Kim, and S.C Joo: A Construction of TMO Object Group Model for Distributed Real-Time Services. The Transaction of Korea Information Science Society, Vol.30, No.5-6 (2003) 307-318
7. Chang-Sun Shin, Chang-Won Jeong, and Su-Chong Joo: Construction of Distributed Object Group Framework and Its Execution Analysis Using Distributed Application Simulation. Lecture Notes in Computer Science, Vol.3207 (2004) 724-733
8. K.H. Kim: Object-Oriented Real-Time Distributed Programming and Middleware. In Proceedings of the 7th International Conference on Parallel and Distributed Systems (2000) 10-20

9. Kim, K.H., Ishida, M., and Liu, J.: An Efficient Middleware Architecture Supporting Time- triggered Message-triggered Objects and an NT-based Implementation. In Proceedings of the IEEE CS 2nd International Symposium on Object-oriented Real-time distributed Computing(ISORC'99) (1999) 54-63
10. S.C. Joo, C.S. Shin, and J.T. Lim: Distributed Programming Developing Tool-Kit Based on Object Group. Program License by Korea Computer Program Deliberation & Mediation Committee (2005) Grant-No.2005-01-172-000228

A Study on Automatic Code Generation Tool from Design Patterns Based on the XMI

Young-Jun Seo and Young-Jae Song

Dept. of Computer Engineering, Kyunghee University,
Sochen-dong, Gihung-gu, Yongin-si, Gyeonggi-do 446-701, Republic of Korea
yjseo@khu.ac.kr, yjsong@khu.ac.kr

Abstract. Recently there have been researches on component reuse for software development productivity and quality improvement, especially on distributed development environment to improve productivity of team development. However, there is a short of schemes to develop and share design components effectively. Therefore, this paper presents a tool that generates a code with compatibility for design pattern to maximize reusability of design component. Presented tool constructs a library that stores explanation information of pattern and structure information of abstract type. Pattern structure information go through the process of instantiation which makes them fit for specific application. Instantiated structure information is generated as a XMI code through code generation template. XMI is supported as a transformed format from most case tools, so it is sure for compatibility.

1 Introduction

Software reuse has been being carried by code unit, in these days, it happens to change by component unit. It is possible for users to expect development productivity and quality improvement by component reuse [1]. Especially, to use software design component for distributed development environment can improve productivity of team development. Also reusability and maintenance ease are expected by unifying these components into an application. However, it is difficult to share software design component in the distributed development environment, so a scheme should be needed to exchange design components internally.

Design pattern has an architecture to solve repeated design problem, and it could be consisted of component and used as a way of communication and understanding [2][3]. Today, some of object oriented modeling tools support design pattern as an architecture element which has interface [4]. But it is possible to make a mistake because most designers apply design patterned architecture to the design by manual work. Also different characteristics should be added each application even though it is the same pattern.

Therefore, this paper presents an automatic code generation tool that generates structure information of design pattern as a XMI base code. Basically, pattern library is constructed for structural ad behavioral pattern information

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 864–872, 2006.

of Gamma. Developer instantiates structure information of pattern by adding application characteristics. Instantiated pattern structure is generated as a XMI code through code generation template. XMI standard is supported as an exchange format in major CASE tool such as Rational Rose of IBM and Together of Borland. Therefore, team developers can have compatibility and easily apply design pattern information in the distributed development environment.

This paper presents as followed. Chapter 2, related work, introduces research trends of pattern related tools and theoretical background on XMI. Chapter 3 explains code generation process and tools' roles which perform each stage of processes. Chapter 4 does comparative analysis with established tools using by qualitative approach. Finally, chapter 5 describes conclusion and further research direction.

2 Related Work

In this chapter, we will look into research trends of tools that are related to patterns and XMI, transformation standard, between UML and XML.

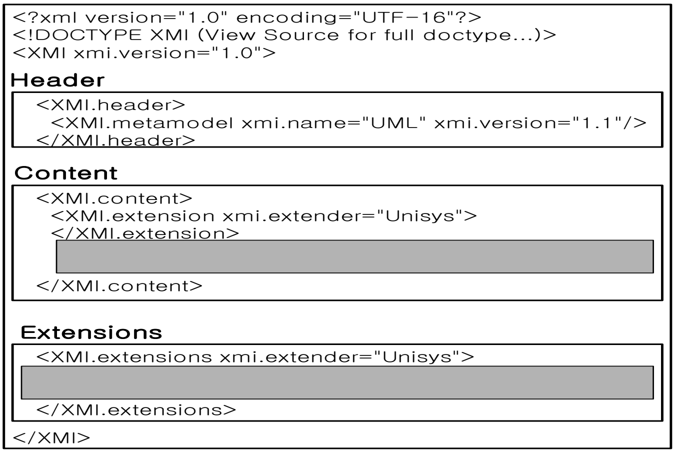
2.1 Research Trends of Tools Related to Patterns

OmniBuiler [5] is a tool which manages the full lifecycle of the application and promotes component reuse and the declarative approach to building applications through the use of Design Patterns. Design patterns are modeled within OmniBuilder as regular objects and can have properties, services, events, methods and behaviors. Design patterns can also be custom programmed by the developer in any object-oriented language.

ModelMaker [6] is a two-way class tree oriented productivity, refactoring and UML-style CASE tool. A number of patterns are implemented as 'ready to use' active agents. A ModelMaker Pattern not only inserts Delphi code fragments to implement a specific pattern, but it also stays 'alive'. Because the pattern is alive, it can reflect changes in the model to the pattern related code or even automatically add or delete members if needed. Many patterns could be expressed using the same class and instance relations. Also many patterns can be implemented very easy using ModelMaker's ability to override methods and keep overridden methods restricted to their origins.

PTL(Pattern Template Library) [7] provides a unified implementation of intrusive data structures and structural patterns in the form of C++ templates. When you need a pattern such as Composite, Flyweight, or a fast Finite State Machine, just grab the class from the library. The library simplifies C++ coding, and is designed as a framework into which users can easily add new patterns.

Budinsky [8]'s research present a tool for generating design pattern code automatically from a small amount of user-supplied information. The tool also gives users an on-line, hypertext rendition of Design Patterns, letting them follow links between patterns instantaneously and search for information quickly. It has three components: The Presenter implements the user interface specified by Presentation Descriptions, which it interprets. The Code Generator generates code that



```
<?xml version="1.0" encoding="UTF-16"?>
<!DOCTYPE XMI (View Source for full doctype...)>
<XMI xmi.version="1.0">
  <XMI.header>
    <XMI.metamodel xmi.name="UML" xmi.version="1.1"/>
  </XMI.header>
  <XMI.content>
    <XMI.extension xmi.extender="Unisys">
    </XMI.extension>
  </XMI.content>
  <XMI.extensions xmi.extender="Unisys">
  </XMI.extensions>
</XMI>
```

Fig. 1. Structure of XMI

implements a pattern. It interprets Code Generation Descriptions, each of which captures how to generate the code for a given pattern. The Mapper specifies how the user interface and code generator components cooperate. It interprets Mapping Descriptions that specify connections and interactions between the other two components.

2.2 XMI(XML Metadata Interchange)

XMI(XML Metadata Interchange) [9] was chosen as the standard to transform UML objected-oriented based into XML web based of the OMG, March, 1999. The structure of XMI is divided into 3 parts as it shown in Figure 1. It writes kinds of meta models transmitted and XMI version information in the Header element and describes meta mode information in the Content element. Finally, it describes present information of meta model in the Extensions element.

3 Automatic Code Generation Tool

3.1 Code Generation Process

Whole code generation process is divided into Pattern Modeling, Pattern Storage, Pattern Instantiation, and Code Generation as shown in Figure 2. In the stage of Pattern Modeling, it inputs pattern explanation and structure information by using Pattern Register. Information input from the former stage is stored in the library by using Pattern Keeper. It shows information that has been stored in the library to developer by using Pattern Presenter in the Pattern Instantiation. Structure information is transformed concrete typed structure to reflect application characteristics. In the Code Generation, it uses Code Generator to generate instantiated structure information into XMI code.

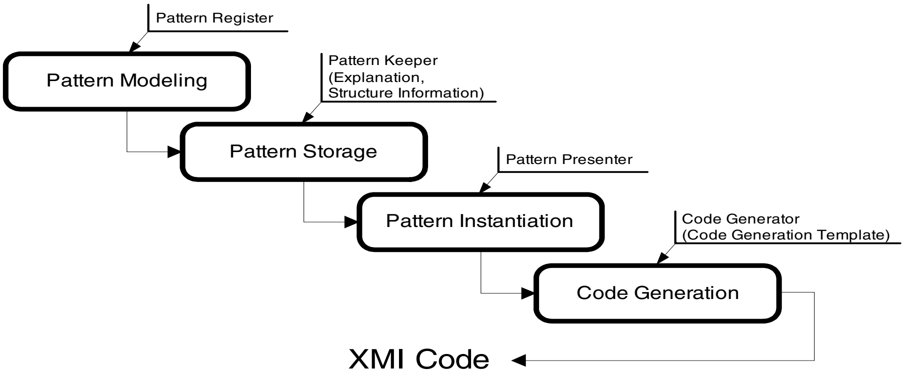
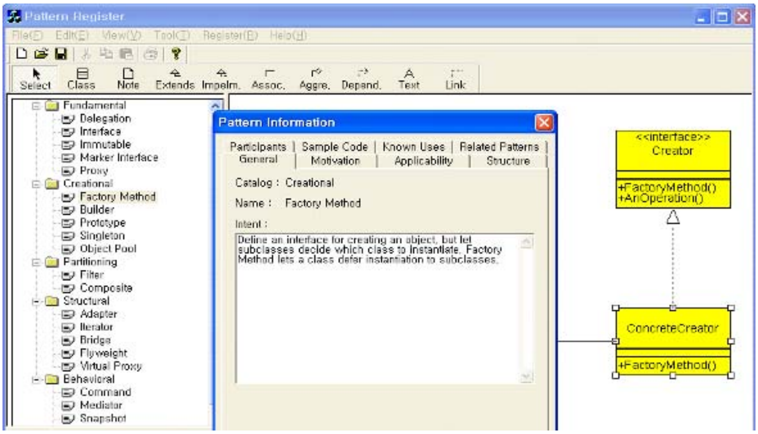


Fig. 2. Code Generation Process



Catalog : Creational

Name : Factory Method

Intent :

Define an interface for creating an object, but let subclasses decide which class to instantiate. Factory Method lets a class defer instantiation to subclasses.

Fig. 3. Pattern Register

3.2 Pattern Register

Pattern Register inputs explanation of design pattern and structure information separately.

Pattern is classified into fundamental, creational, partitioning, structural, and behavioral pattern according to purposes. Having done that, they are divided into sections such as intent, motivation, structure, participants etc and input. Structure information that is belonged to structure section will be modeling by using UML notion. Figure 3 is a picture that displays a form of Pattern Register.

3.3 Pattern Keeper

Pattern Keeper stores explanation and structure information that are input from Pattern Register into the Pattern Library. Figure 4 is a picture that displays

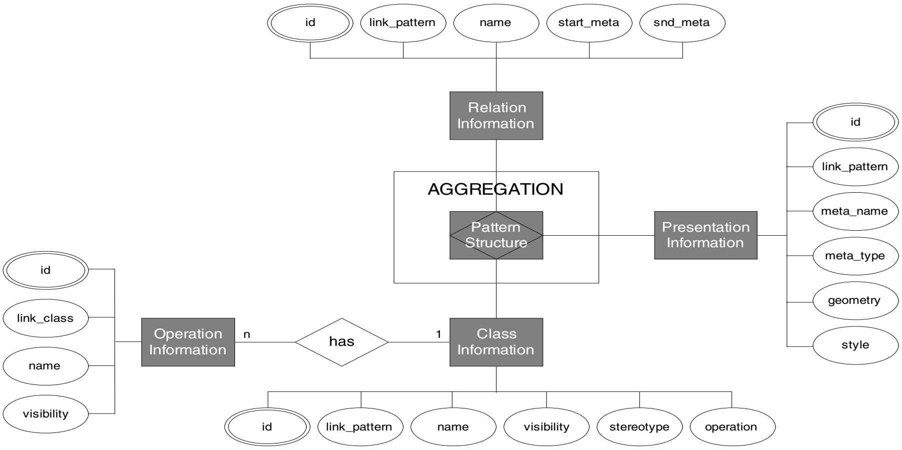


Fig. 4. Storage Structure of Pattern

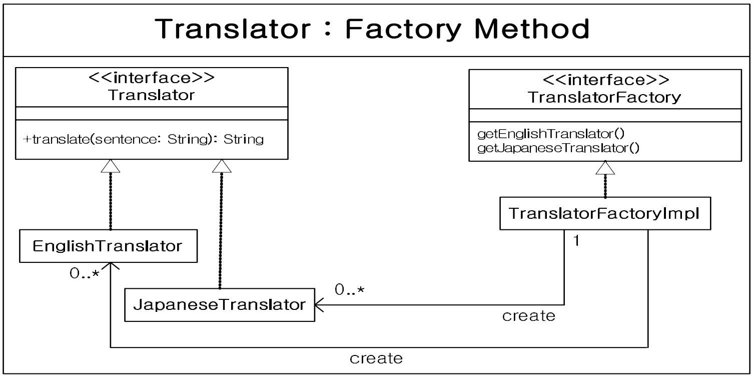


Fig. 5. Instantiation of Factory Method Pattern

structure information as ERM(Entity Relationship Model). Pattern Structure is consisted of the mixture of Class Information, Relation Information, and Presentation Information entity. Class Information entity has class name, visibility, and stereotype attribute. It forms a multiplicity relationship of 1:n with Operation Information entity. Relation Information entity has relation information between classes or interfaces. And it is consisted of relation name, start class(interface) id, and end class(interface) id. Relation Information entity is consisted of class (interface), geometry of relation information entity and style information.

3.4 Pattern Presenter

Presenter sends each of explanation information and structure information that are retrieved from the pattern library to pattern browser and pattern instantiater to display related information.

Pattern instantiater transforms structure information of abstract type into structure of concrete type which reflects application characteristics [10]. Transformation process is that followed. The first, it renames the names of class and operations referring name allocation table. Secondly, if there is a new born class, it specifies relation with established class or interface. Thirdly, if operation is defined in the abstract class of interface, the class that is related to inheritance overrides applicable operation. Figure 5 is a picture that Factory Method pattern is transformed suitable to translator system.

3.5 Code Generator

Code Generator generates XMI code by mapping meta model of applicable pattern on code generation template. Listing 1 shows the structure of code generation template that generates XMI code. Each code segment will be mapped by parameter surrounding #, after that, they will extend to recursive.

```
[code GEN_CODE]
<XMI xmi.version="1.0">
#HEADER#
#CONTENT#
#EXTENSIONS#
</XMI>
[/code GEN_CODE]

//////////////////////////////////////////////////////////
// HEADER code segment
//////////////////////////////////////////////////////////
[code HEADER]
        <XMI.header>
                <XMI.documentation>
                <XMI.exporter>Code Generator</XMI.exporter>
                <XMI.documentation>
                <XMI.metamodel xmi.name="UML" xmi.version="1.1"/>
        </XMI.header>
[/code HEADER]

//////////////////////////////////////////////////////////
// CONTENT code segment
//////////////////////////////////////////////////////////
[code CONTENT]
<XMI.content>
        <name>#MODEL_NAME#</name>
        <ownedElement>
        [repeat CLASS]
                #CLASS#
        [/repeat CLASS]
        [repeat RELATION]
            #RELATION#
        [/repeat RELATION]
        </ownedElement>
[/code CONTENT]

[code CLASS]
        <Class xmi.id="#ID#">
                <name>#CLASS_NAME#</name>
                <feature>
                        [repeat OPERATION]
                                #OPERATION#
                        [/repeat OPERATION]
                </feature>
        <Class>
[/code CLASS]

[code OPERATION]
        <Operation xmi.id="#ID#">
                <name>#OPER_NAME#</name>
                <visibility xmi.value="#VISIBILITY#"/>
        </Operation>
[/code OPERATION]

[code RELATION]
        <Association xmi.idref="#ID#">
                <name>#ASSOC_NAME#</name>
                <connection> .. </connection>
</Association>
[/code RELATION]
```

Listing.1. Structure of Code Generation Template

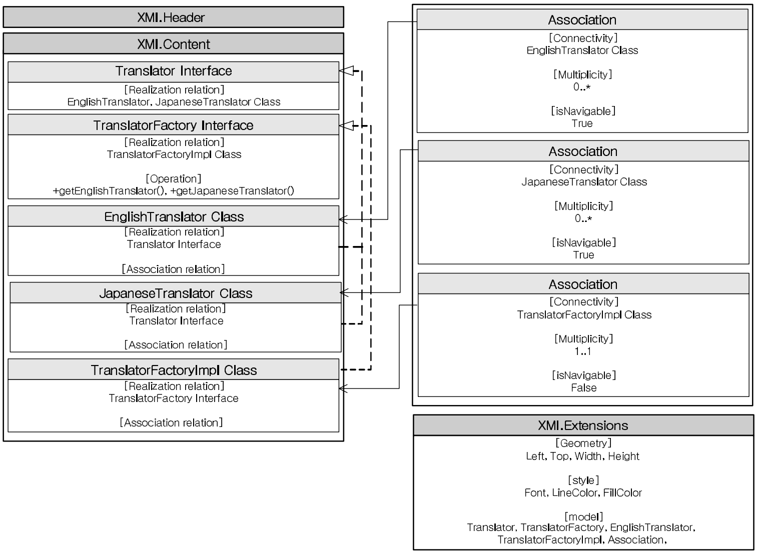


Fig. 6. XMI Code Structure of Factory Method Pattern

For instance, Factory Method pattern in Figure 5 can be generated as XMI code like in Figure 6.

<XMI Content> element describes meta models of Factory Method pattern, that is, meta data that are about Translator, TranslatorFactory interface, EnglishTranslator, JapaneseTranslator, TranslatorFactoryImpl class, Realization, and Association relation. <XMI.Extensions> element describes presentation, that is, geometry and style, about each meta models that have been described in <XMI.Content>. Connected arrow between Translator interface and EnglishTranslator(JapaneseTranslator) class shows a relation between two objects. And it is the same between TranslatorFactory interface and TranslatorFactoryImpl class, whereas Association relation is shown there between

two objects by putting Association element separately between EnglishTranslator(JapaneseTranslator) class and TranslatorFactoryImpl class.

4 Comparison Analysis

I have made a comparison and evaluation between established tools like PTL [7] of Code Farms and DPL [8] of IBM and tools presented this paper. They are shown in the Table 1, and there are some features as followed.

Table 1. Comparison Table

	Pattern Template Library	Design Pattern Library	Proposed Tool
Possiblity of Pattern Expansion	○	×	○
Compatibility of Structure	×	×	○
Support of Structure Modeling	△	×	○
Web based Environment	×	○	×

○: full support, △: partial support, ×: non-support

The first, Possibility of Pattern Expansion. Most tools includes pattern of Gamma basically. PTL is able to add a data structure that derives a pattern apart from pattern of Gamma. And also it can add user pattern in the tool presented, if needed. However, only DPL cannot propose a method that can add pattern. The second, Compatibility of Structure. Pattern structure code that is generated from established tools is not able to provide compatibility with CASE tool or platform independency. However, code generated from tool presented can be provided from CASE tool supporting XMI regardless of platform. The third, Support of Structure Modeling. It provides a method that is able to do pattern structure modeling and instantiation by UML which is used by only presented tool. The last, Web based Environment. DPL is operated in the web based environment, so it can be used with no installation.

5 Conclusion

This paper has introduced how to develop software design component to the form with compatibility and can be shared to team developers. Presented tool builds a library that stores structure information of abstract type with explanation information of pattern. Structure information of Pattern goes through the process that does instantiation which is suitable for specific application. Instantiated structure information is generated XMI code via code generation template.

Pattern structure generated XMI code shows a difference from established tools in the point of that has a compatibility with most tools which support XMI. That is, advantages of design pattern, software productivity and quality improvement, could be reflected to software development directly. And reusability and maintenance ease could be expected through reuse of pattern structure.

For further research direction, to expend code generation template is the main task in order to generate code by another XMI transformation code like UXF(UML eXchange Format) [11] apart from XMI.

References

1. S. Yau, N. Dong, “Integration in Component-Based Software Development Using Design Patterns”, Proc. Of the 24th COMPSAC, (2000)
2. E. Gamma, R. Helm, R. Johnson and J.Vlissides, “Design Patterns: Elements of Reusable Object-Oriented Software”, Addison-Wesley, (1995)
3. Mark Grand, “Patterns in Java”, Vol. 1, Wiley, (1998)
4. S. Yacoub, H. Xue, H. Ammar, “Automating the Development of Pattern-Oriented Designs for Application Specific Software System”, Proc. Of the 3rd ASSET, (2000)
5. OmniBuider, Available online: http://www.omnibuider.com/overview/design.htm
6. ModelMaker, Available online: http://www.modelmakertools.com
7. Code Farms, Available online: http://www.codefarms.com
8. F.J. Budinsky, M.A. Finnie, J.M. Vlissides, P.S. Yu, “Automated code generation from design patterns”, IBM Systems Journal, Vol. 35, No. 2, (1996)
9. OMG, XML MetaData Interchange(XMI) 1.0, Available online: http://www.omg.org, (1999)
10. M. Ohtsuki, N. Yoshida, “A Source Code Generation Support System Using Design Pattern Documents Based on SGML”, Proc. Of the APSEC, (1998)
11. J. Suzuki, Y. Yamamoto, “Managing the Software Design Documents with XML”, http://www.yy.ics.keio.ac.kr/ suzuki, (1999)

Design of Opportunity Tree for Organization's Process Strategy Decision-Making Based on SPICE Assessment Experience

Ki Won Song[1], Haeng Kon Kim[2], and Kyung Whan Lee[1]

[1] School of Computer Science and Engineering, Chung-Ang University, Korea
{kwsong, kwlee}@object.cau.ac.kr
[2] School of Computer Science and Engineering, Catholic University of Daegu, Korea
{hangkon}@cuth.catagegu.ac.kr

Abstract. With growing interest in Software Process Improvement (SPI), many companies are introducing international process models and standards. For efficient process improvement, work performance should be enhanced in line with organization's vision by identifying areas for improvement and risks with process assessment standards such as SPICE or CMMI and then mapping them in the software development environment.. SPICE(Software Process Improvement and Capability dEtermination ISO/IEC 15504) is most widely used process assessment model in the SPI work today.

This paper expands the number of routes for improvement in the existing OTF model from 8 to 24 and the number of processes in SEF model from 40 of the TR version to 51 of the IS version. This paper proposes OTEM (Opportunity Tree Enterprise Model) which can provide optimal strategies for the organization's vision by using Balanced Scorecard method to determine optimal strategies to the expanded routes for improvement.

1 Introduction

Many organizations, including government agencies, define enterprise-wide measures to reflect the relative health of their organization. These measures help guide an organization's overall performance and process improvement effort [1].

For effective process improvement of an organization, it is most important to accurately evaluate the organization's project performance first. Unfortunately, IT-based organizations are very difficult to measure its performance due to invisibility and variability of software. Lynch and Cross suggested Performance Pyramid to overcome this difficulty. The PP (Performance Pyramid) is a method to measure project performance which is focused on 3 targets of project performing, which are customers' satisfaction, flexibility, and productivity, and selects quality, delivery, cycle time, and waste as performance attributes.

Based on this PP model, this paper measures quality and delivery attributes for external effectiveness of organization and cycle time and waste attributes for internal efficiency by making questionnaires based on GQM (Goal Question Metrics) and

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 873 – 882, 2006.

inputting the analysis results to PPM (Project Performance Measure) model. The determined performance and qualitative vision of the organization is mapped onto the 24 routes for improvement and 9 categories (51 processes), using the Balanced Scorecard and then, priority is determined among them. SEF suggests optimal improvement strategies for the organization by statistically analyzing the SPICE assessment results.

By statistically analyzing merits and demerits by level from the results and SPICE assessment results of 52 appraisals and 497 processes which were assessed by KASPA (Korea Association of Software Process Assessors) from 1999 to 2005, this paper derives weakness for improvement by level and suggests a framework of opportunity tree model to propose an optimal improvement strategy for reaching each level of SPICE and uses of it [7][10].

2 Basic Study

2.1 The Balanced Scorecard Framework

The balanced scorecard is an industry-recognized best practice for measuring the health of an organization. It can be used as a management tool for translating an organization's mission and strategic goals into a comprehensive set of performance measures that provide the framework for an enterprise measurement and management system [1].

The balanced scorecard methodology is based on four perspectives of an organization's performance—customer, financial, internal process, and learning and growth. These perspectives are illustrated in <Figure 1> [1].

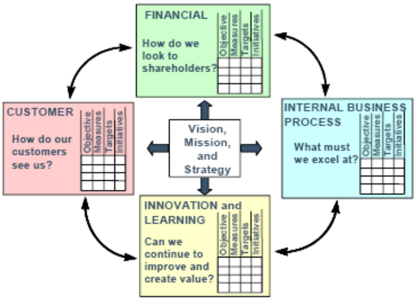


Fig. 1. The Balanced Scorecard Framework

2.2 GQM (Goal-Question-Metrics) Procedures

GQM process is a series of procedures as follows: Set an organization's goals through GQM approach; Set goals of project in each area.

Make questions and develop metrics; Measure accomplishment of the goals using the metrics [3][12].

As shown in <Figure 2>, GQM process is generally composed of vision, objectives, and areas belonging to external effectiveness or internal efficiency [8].

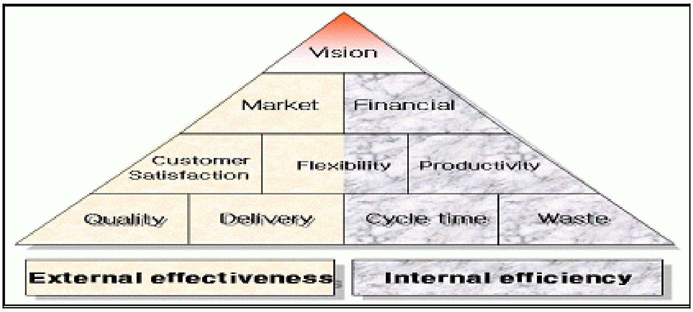


Fig. 2. Lynch and Cross's Performance Pyramid

For example, if an organization's vision is improvement of public recognition on it, it should concentrate on market shares rather than on financial performance, and put its priority on customer satisfaction and flexibility rather than on flexibility and productivity [2][6]. In order to satisfy customers, quality and delivery should be more emphasized than cycle time and waste.

Quality and delivery is goals to measure external effectiveness while cycle time and waste is to measure internal efficiency [5].

Here, goals are set again in each area, strategies for process improvement are developed through GQM approach, and measurement is carried out [8][13].

2.3 GQM (Goal-Question-Metrics) Approach

GQM approach involves three steps.

First is conceptual step. It consists of elements such as object, purpose, viewpoint and focus. In this step, major goal are set.

Second is operational step. In this step, questions are derived from the goals that must be answered in order to determine if the goals have been achieved. It asks questions about how the specific goals have been assessed or achieved.

Third is quantitative step in which proper answers are given to the questions[5].

Through the three steps, a metrics system is made. These metrics can be used as a measurement tool which metrics a set of data is associated with every question in order to answer it in a quantitative way[6] [8] [9] [11].

3 Organization's Project Performance Measure

This section suggests PPM (Project Performance Measure) model to quantitatively measure organization's project performance. The PPM model calculates project performance with GQM approach in terms of 4 performance attributes of the Performance Pyramid developed by Lynch and Cross.

GQM (Goal-Based Question Metric) process is to measure it by establishing organization's objectives, developing project goals in each field in compliance with the objectives, asking questions to deal with them, and developing a metric.

Based on the performance pyramid, the author made quantitative GQM questionnaires with GQM approach to measure performance of an organization and calculate the earned value.

A quantitative GQM questionnaire is composed of questions to measure the extent to which external effectiveness and internal efficiency of an organization have reached the goals. In each area, a project goal is set and strategies for project process improvement are determined and measured with GQM method.

For each of the 4 goals, 2 for quality and delivery to measure external effectiveness of an organization, and 2 for cycle time and waste to measure internal efficiency, questions and metrics are made with GQM method.

3.1 GQM Quantitative Questionnaire from Meta Data

This section proposes GQM quantitative questionnaire made from general meta data. Procedures for making the questionnaire involve three steps: set goals; give questions; gain metrics.

First is conceptual step. It consists of elements such as object, purpose, viewpoint and focus. In this step, major goal are set. Second is operational step. In this step, questions are derived from the goals. Third is quantitative step in which proper answers are given to the questions. Through the three steps, a metrics system is made. Twenty measurable metrics were made for eight questions.

3.2 Project Performance Measurement Model

This section proposes PPM to calculate project performance of an organization in terms of external effectiveness and internal efficiency. See reference [8] of this paper for metrics elements and measure method. The calculation forms to measure a project performance of an organization in terms of external effectiveness and internal efficiency for 4 goals are shown in <Table 1>.

Table 1. Calculation forms to measure a project performance of an organization in terms of external effectiveness and internal efficiency

External effectiveness PPM(qd) = (PPM(q)+PPM(d)/2	PPM(q): quality effectiveness score	$\sum((100 - each\, defect\, rate) + defect\, management\, rate)/4$
	PPM(d): schedule effectiveness score	$\sum(100 - on\, schedule\, rate\, at\, each\, stage)/5$
Internal efficiency PPM(c,w)=(PPM(c))+PPM(w)/2	PPM(c): effort efficiency score	$\sum(effort\, correspndence\, rate\, at\, each\, stage)/4$
	PPM(w): resource efficiency score	$\sum(all\ factors) - (2 \times (100 - [rework\ per\, code])/6$

But to select the optimal process improvement strategy, the organization's vision and past experience of the optimal project improvement should be reflected.

Therefore, this paper proposes SEF(SPICE Experience Factory) and OTEM (Opportunity Tree Enterprise Model) which can reflect SPICE experiences of process improvement and organization's vision.

3.3 DB Construction for SEF (SPICE Experience Factory)

Based on the results of SPICE assessments conducted from 1999 to 2005 by KASPA, this section suggests SEF (SPICE Experience Factory) model which can present effective process improvement items (root words) from experience data on process improvement and project performance results measured with PPM model.

SEF Model Design Procedures

1. Collect data on SPICE assessment results
2. Classify general strengths and weakness for improvement, specific strengths and weakness for improvement from the collected data by process and by level, and then derive root works from each item
3. Select meta data from the derived root word
4. Design internal schema and construct DB

This section describes the outline of the DB constructed according to the SEF model design procedure above. The DB is designed to satisfy the following three conditions. First, it should be scalable so as to add new assessment results.

Second, it should provide necessary information to SEF (SPICE Experience Factory) model. Third, its confidentiality should be secured to protect information of the assessed organization. The DB satisfying the conditions above is composed of 4 tables as seen in <Table 2>.

Table 2. Composition of SEF DB

Table Name	Description
Assessment Info	Manages SPICE assessment information (assessed OU, assessment period, assessment scope, assessment version International Standard)
Process Profile	Saves rating results and accomplished levels by process
Result Data	Saves assessment results (Generally general strengths/ weakness for improvement, strengths / weakness for improvement for each process) and derived root word
Root Metadata	Saves and Manages root word and relevant category
PPM Score	PPM scores by SPICE level

3.4 DB Design

For accumulation and utilization of SPICE assessment experiences, this paper describes the DB to reuse efficiently the analyzed assessment results.

The DB was designed to satisfy following three conditions. First, it should be scalable to add new assessment results. Second, it should provide necessary information

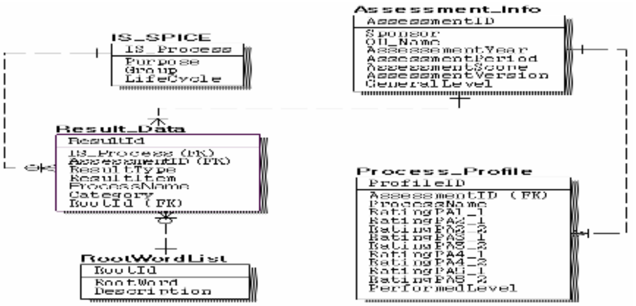


Fig. 3. Composition of DB Schema

to SEF (SPICE Experience Factory) model. Third, its confidentiality should be kept to protect information of the assessed organization. The composition of DB schema representing structure of each table is shown in <Figure 3>.

4 OTEM (Opportunity Tree Enterprise Model)

This section proposes OTEM which can determine an effective process improvement route which is optimized for an organization's vision, based on PPM results and SEF experience data.

The composition and use-case of OTEM and OTF Model are illustrated in <Figure 4> and <Figure 5> respectively.

4.1 Qualitative Questionnaire

Qualitative questionnaire is designed to identify "vision weight" for each focus. Vision weight here reflects the extent to which the organization pursues the vision. Qualitative questionnaire consists of three steps as shown in <Table 4>.

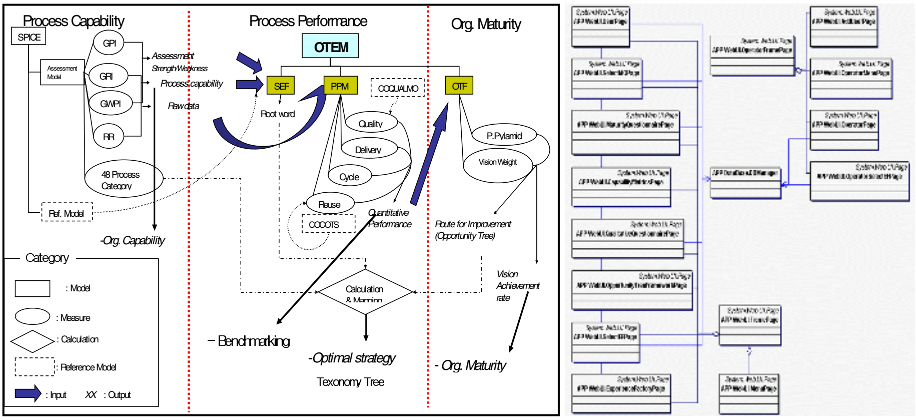
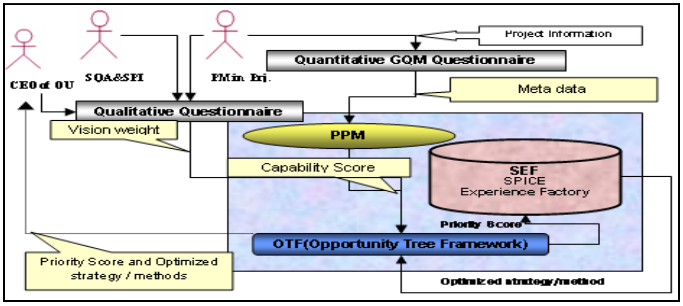


Fig. 4. Composition of OTEM design



CEO of OU
SQA&SPI
PM in Eni.
Project Information
Quantitative GQM Questionnaire
Meta data
Qualitative Questionnaire
Vision weight
PPM
Capability Score
SEF
SPICE
Experience Factory
Priority Score
OTF(Opportunity Tree Framework)
Priority Score and Optimized strategy / methods
Optimized strategy/method

Fig. 5. Use-case of OTEM

Table 3. Decision-making Procedure to determine priority for improvement of OTEM

1. Measure project performance through PPM model
2. Input vision weight through qualitative questionnaire
3. Calculate priority scores of 8 improvement items
4. Input the calculated priority and PPM performance score in SEF
5. Search the optimal strategy or method from SEF (SEF responds improvement items consisting of root words which are prioritized by process and by level.)

Table 4. Structure of Qualitative Questionnaire

Step 1: Input scores of visions of market share and financial performance by percentage from the point of view of business unit.
Step 2: Input scores of visions of customer satisfaction, flexibility, and productivity by percentage from the point of view of core business process.
Step 3: Input scores of visions of quality, delivery, cycle time, and waste by percentage from the point of view of development group.

Like this, by obtaining viewpoints of each stakeholder through the qualitative questionnaire, an optimal improvement strategy can be developed which reflect all viewpoints of them.

4.2 Routes for Improvement

This section proposes a model to find optimal routes for improvement. The model integrates project performance calculated through PPM and vision weight of stakeholders obtained through qualitative questionnaire. OTF model is used to determine priority for improvement and performance scores to accomplish improvements in compliance with the organization's vision, which is impossible with simple calculation alone through PPM.

The existing 8 routes of OTF, however, were not adequate. So, this paper expands the number of them to 24 as shown in <Table 5>.

Table 5. Route for improvement

Route for improvement [Ri]	Route for improvement [Ri]
1. MCQ = [M]+[C]+[Q]score	13. MCC = [M]+[C]+[C]score
2. MFQ = [M]+[F]+[Q]score	14. MFC = [M]+[F]+[C]score
3. MPQ = [M]+[P]+[Q]score	15. MPC = [M]+[P]+[C]score
4. FCQ = [F]+[C]+[Q]score	16. FCC = [F]+[C]+[C]score
5. FFQ = [F]+[F]+[Q]score	17. FFC = [F]+[F]+[C]score
6. FPQ = [F]+[P]+[Q]score	18. FPC = [F]+[P]+[C]score
7. MCD = [M]+[C]+[D]score	19. MCC = [M]+[C]+[C]score
8. MFD = [M]+[F]+[D]score	20. MFC = [M]+[F]+[C]score
9. MPD = [M]+[P]+[D]score	21. MPC = [M]+[P]+[C]score
10. FCD = [F]+[C]+[D]score	22. FCC = [F]+[C]+[C]score
11. FFD = [F]+[F]+[D]score	23. FFC = [F]+[F]+[C]score
12. FPD = [F]+[P]+[D]score	24. FPC = [F]+[P]+[C]score

4.3 Model to Draw Routs for Improvement Based on BSC Method

This section proposes a method of mapping which applies Balanced Scorecard Framework in order to find optimal routes of improvement for software development companies. What we call BSC (Balanced Scorecard) is a method to determine future corporate value by adding perspectives of customer, internal business process, innovation and learning to the existing performance measurement system used in Business Administration field which emphasizes only financial perspective.

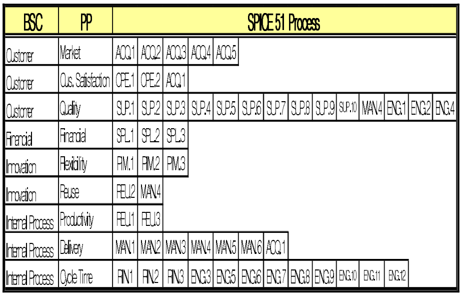
This paper maps the 4 perspectives onto 9 categories of PP(Performance Pyramid) and in turn, onto 9 categories of SPICE process so as to identify optimal routes of improvement.

Composition and results of mapping table are shown in <Table 6>.

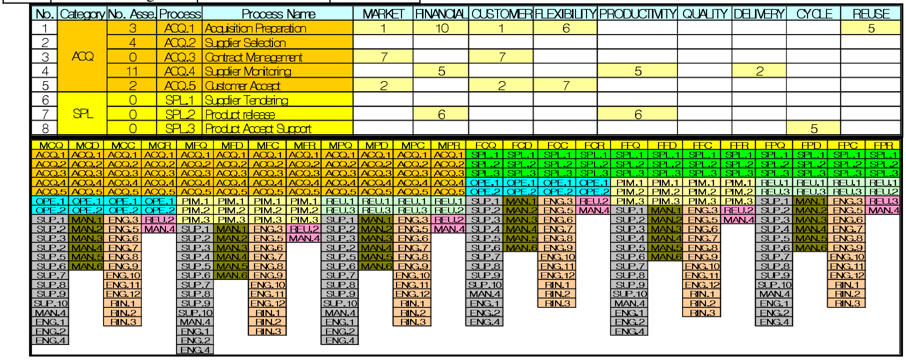
The mapping process is determined according to weight of correlation among the attributes.

Table 6. Calculation method of route for improvement

No	BSC Focus	PP Category	SPICE Category
1	Customer	Market	ACQ
2	Financial	Financial	SPL
3	Customer	Customer Satisfaction	ENG
4	Innovation and Learning	Flexibility	OPE
5	Internal Process	Productivity	SUP
6	Customer	Quality	MAN
7	Internal Process	Delivery	PIM
8	Internal Process	Cycle time	RIN
9	Innovation and Learning	Reuse	REU



BSC	PP	SPICE 51 Process													
Customer	Market	ACQ.1	ACQ.2	ACQ.3	ACQ.4	ACQ.5									
Customer	Cus. Satisfaction	OPE.1	OPE.2	ACQ.1											
Customer	Quality	SUP.1	SUP.2	SUP.3	SUP.4	SUP.5	SUP.6	SUP.7	SUP.8	SUP.9	SUP.10	MAN.4	ENG.1	ENG.2	ENG.4
Financial	Financial	SPL.1	SPL.2	SPL.3											
Innovation	Flexibility	PIM.1	PIM.2	PIM.3											
Innovation	Reuse	REU.2	MAN.4												
Internal Process	Productivity	REU.1	REU.3												
Internal Process	Delivery	MAN.1	MAN.2	MAN.3	MAN.4	MAN.5	MAN.6	ACQ.1							
Internal Process	Cycle Time	RIN.1	RIN.2	RIN.3	ENG.3	ENG.5	ENG.6	ENG.7	ENG.8	ENG.9	ENG.10	ENG.11	ENG.12		



No.	Category	No. Asse.	Process	Process Name	MARKET	FINANCIAL	CUSTOMER	FLEXIBILITY	PRODUCTIVITY	QUALITY	DELIVERY	CYCLE	REUSE
1	ACQ	3	ACQ.1	Acquisition Preparation	1	10	1	6					5
2		4	ACQ.2	Supplier Selection									
3		0	ACQ.3	Contract Management	7		7						
4		11	ACQ.4	Supplier Monitoring		5			5		2		
5		2	ACQ.5	Customer Accept	2		2	7					
6	SPL	0	SPL.1	Supplier Tendering									
7		0	SPL.2	Product release		6			6				
8		0	SPL.3	Product Accept Support								5	

MCQ	MCD	MCC	MCR	MFQ	MFD	MFC	MFR	MPQ	MPD	MPC	MPR
ACQ.1	ACQ.1	ACQ.1	ACQ.1	ACQ.1	ACQ.1	ACQ.1	ACQ.1	ACQ.1	ACQ.1	ACQ.1	ACQ.1
ACQ.2	ACQ.2	ACQ.2	ACQ.2	ACQ.2	ACQ.2	ACQ.2	ACQ.2	ACQ.2	ACQ.2	ACQ.2	ACQ.2
ACQ.3	ACQ.3	ACQ.3	ACQ.3	ACQ.3	ACQ.3	ACQ.3	ACQ.3	ACQ.3	ACQ.3	ACQ.3	ACQ.3
ACQ.4	ACQ.4	ACQ.4	ACQ.4	ACQ.4	ACQ.4	ACQ.4	ACQ.4	ACQ.4	ACQ.4	ACQ.4	ACQ.4
ACQ.5	ACQ.5	ACQ.5	ACQ.5	ACQ.5	ACQ.5	ACQ.5	ACQ.5	ACQ.5	ACQ.5	ACQ.5	ACQ.5
OPE.1	OPE.1	OPE.1	OPE.1	PIM.1	PIM.1	PIM.1	PIM.1	REU.1	REU.1	REU.1	REU.1
OPE.2	OPE.2	OPE.2	OPE.2	PIM.2	PIM.2	PIM.2	PIM.2	REU.3	REU.3	REU.3	REU.2
SUP.1	MAN.1	ENG.3	REU.2	PIM.3	PIM.3	PIM.3	PIM.3	SUP.1	MAN.1	ENG.3	REU.2
SUP.2	MAN.2	ENG.5	MAN.4	SUP.1	MAN.1	ENG.3	REU.2	SUP.2	MAN.2	ENG.5	MAN.4
SUP.3	MAN.3	ENG.6		SUP.2	MAN.2	ENG.5	MAN.4	SUP.3	MAN.3	ENG.6	
SUP.2	MAN.4	ENG.7		SUP.3	MAN.3	ENG.6		SUP.2	MAN.4	ENG.7	
SUP.5	MAN.5	ENG.8		SUP.4	MAN.4	ENG.7		SUP.5	MAN.5	ENG.8	
SUP.6	MAN.6	ENG.9		SUP.5	MAN.5	ENG.8		SUP.6	MAN.6	ENG.9	
SUP.7		ENG.10		SUP.6	MAN.6	ENG.9		SUP.7		ENG.10	
SUP.8		ENG.11		SUP.7		ENG.10		SUP.8		ENG.11	
SUP.9		ENG.12		SUP.8		ENG.11		SUP.9		ENG.12	
SUP.10		RIN.1		SUP.9		ENG.12		SUP.10		RIN.1	
MAN.4		RIN.2		SUP.10		RIN.1		MAN.4		RIN.2	
ENG.1		RIN.3		MAN.4		RIN.2		ENG.1		RIN.3	
ENG.2				ENG.1		RIN.3		ENG.2			
ENG.4				ENG.2				ENG.4			
				ENG.4							

FCQ	FCD	FCC	FCR	FFQ	FFD	FFC	FFR	FPQ	FPD	FPC	FPR
SPL.1	SPL.1	SPL.1	SPL.1	SPL.1	SPL.1	SPL.1	SPL.1	SPL.1	SPL.1	SPL.1	SPL.1
SPL.2	SPL.2	SPL.2	SPL.2	SPL.2	SPL.2	SPL.2	SPL.2	SPL.2	SPL.2	SPL.2	SPL.2
SPL.3	SPL.3	SPL.3	SPL.3	SPL.3	SPL.3	SPL.3	SPL.3	SPL.3	SPL.3	SPL.3	SPL.3
OPE.1	OPE.1	OPE.1	OPE.1	PIM.1	PIM.1	PIM.1	PIM.1	REU.1	REU.1	REU.1	REU.1
OPE.2	OPE.2	OPE.2	OPE.2	PIM.2	PIM.2	PIM.2	PIM.2	REU.3	REU.3	REU.3	REU.2
SUP.1	MAN.1	ENG.3	REU.2	PIM.3	PIM.3	PIM.3	PIM.3	SUP.1	MAN.1	ENG.3	REU.3
SUP.2	MAN.2	ENG.5	MAN.4	SUP.1	MAN.1	ENG.3	REU.2	SUP.2	MAN.2	ENG.5	MAN.4
SUP.3	MAN.3	ENG.6		SUP.2	MAN.2	ENG.5	MAN.4	SUP.3	MAN.3	ENG.6	
SUP.4	MAN.4	ENG.7		SUP.3	MAN.3	ENG.6		SUP.4	MAN.4	ENG.7	
SUP.5	MAN.5	ENG.8		SUP.4	MAN.4	ENG.7		SUP.5	MAN.5	ENG.8	
SUP.6	MAN.6	ENG.9		SUP.5	MAN.5	ENG.8		SUP.6	MAN.6	ENG.9	
SUP.7		ENG.10		SUP.6	MAN.6	ENG.9		SUP.7		ENG.10	
SUP.8		ENG.11		SUP.7		ENG.10		SUP.8		ENG.11	
SUP.9		ENG.12		SUP.8		ENG.11		SUP.9		ENG.12	
SUP.10		RIN.1		SUP.9		ENG.12		SUP.10		RIN.1	
MAN.4		RIN.2		SUP.10		RIN.1		MAN.4		RIN.2	
ENG.1		RIN.3		MAN.4		RIN.2		ENG.1		RIN.3	
ENG.2				ENG.1		RIN.3		ENG.2			
ENG.4				ENG.2				ENG.4			
				ENG.4							

The calculation results using vision weight from qualitative questionnaire and project performance scores can be seen in <Figure 6>.

In <Figure 4>, P is the arithmetic score of the extent of project achievement when same weight is given to each vision. W is the weight value gained through qualitative questionnaire. CP is the value corrected by applying vision weight to the project achievement scores. CW is the value corrected by reflecting visions of upper level.

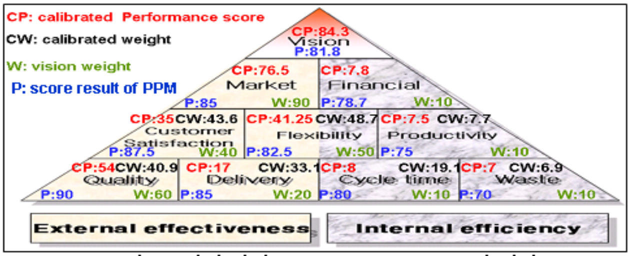


Fig. 6. Calculation of Vision Achievement

Once the corrected value is calculated, an algorithm should be proposed to select the most appropriate route for improvement, using this value. By analyzing gap between current project performance and the organization's vision through this algorithm, the optimal route can be determined. The algorithm calculation method to find optimal routes for improvement is presented in <Table 7>.

Table 7. Calculation method of route for improvement

	Calculation method
Route For improvement	24 Route for Improvement
Vision Achievement	Vision score of [business unit + Core business process +department group]
Arithmetic Score	Arithmetic score of [business unit+Core business process+ department group]
Vision Score	Calibrated vision score of [business unit+Core business process+ department group]
Priority of improvement	100-(Arithmetic score/300)*100

The arithmetic score means the current performance score of the organization in selecting routes for process improvement.

Here, items with lower scores mean their performance scores are also lower, so need more effort for improvement. If the vision score is higher, it can be considered that the organization attaches more importance to it.

5 Conclusion and Hereafter Research

Criteria for selecting routes for improvement with OTEM were determined by reflecting following three results.

1) Deriving objectives of all stakeholders and project performance of organization.
2) Mapping the BSC onto the Performance Pyramid and 51 processes of SPICE.
3) Deriving priority of areas of weakness for improvement.

The priority of routes for improvement is determined by calculating both cases of arithmetically low scores and high scores in terms of importance.

Stakeholders' objectives are focused on the goal of improving organization's project performance in achieving its visions in terms of market and financial benefit.

Scores of 4 work performance units (Quality, Delivery, Cycle time, and Waste) are calculated using PPM. Based on the scores, the priority of 24 routes for improvement is output from OTF. When inputting this result to SEF, root words of issues for process improvement are shown based on SPICE assessment results.

The root words of issues for process improvement can develop the optimal criteria of the organization.

From the results of this paper, following two advantages can be obtained.

First, quantitative project performance of an organization can be measured using PPM model before it works to improve processes in earnest.

Second, the optimal criteria for process improvement can be developed by deriving areas for improvement with OTF model and SEF model based on current performance of the organization, priority of improvement strategies which reflect stakeholders' vision with OTF model and SEF and SPICE assessment data.

Finally, more detail strategies can be derived because it expands the number of routes of improvement in OTF to 24 and the number of processes of SEF from 40 to 51.

In the future, by collecting actual data through the OTEM prototype (http://otem.ksapa.org) which has been implemented with ASP .Net now, if the number of cases of actual data collection on PPM projects exceeds 30, data reliability can be analyzed on assumption of F-distribution because data of all models which can be analyzed by F-distribution shows normal distribution. In this case, 4 performance units need practical reliability verification through experiential case study by implementing web-based tool using each typical performance.

References

[1] Wolfhart Goethert, "Matt Fisher Deriving Enterprise-Based Measures Using the Balanced Scorecard and Goal-Driven Measurement Techniques", SEI Technical Note CMU/SEI-2003-TN-024, October 2003

[2] Richard L. Lynch, Kelvin F. Cross, "Measure up!", Blackwell, 1995.

[3] Williams A. Florac, Anita D. Carleton, "Measuring the software process", SEI, Addison Wesley, 1999.

[4] Kyung whan Lee, "Quantitative Analysis for SPI", Corporation seminar, Feb. 17. 2003.

[5] Frank Van Latum, Rini Van Soligen, "Adopting GQM-Based Measurement in an industrial Environment", IEEE software, 1998.

[6] V. R. Basili, G. Caldiera, H. D. Rombach, "Goal Question Metric Paradigm", Encyclopedia of Software Engineering, John Wiley & Sons, Volume 1, pp. 528–532, 1994

[7] KSPICE (Korea Association of Software process Assessors), SPICE Assessment Report http://kaspa.org, 2002–2004

[8] Ki-Won Song, " Design of Opportunity Tree Framework for Effective Process Improvement based on Quantitative Project Performance", SERA05, CMU, 2005

[9] Kyung whan Lee, "ROI of IT Business", The Federation of Korean Information Industries, May 2003.

[10] ISO/IEC JTC1/SC7 15504: Information Technology-Software Process Assessment, ISO, ver.3.3, 1998

[11] Tim Kasse, "Action Focused Assessment for software process improvement", Artech House, 2002.

[12] Barry Boehm, "Value-Based Software Engineering: Case Study", IEEE Computer, pp. 33–41, 2003.

[13] Donald J. Reifer, "Making the Software Business Case", Addison-Wesley, 2002.

SUALPPA Scheme: Enhanced Solution for User Authentication in the GSM System

Mi-Og Park[1] and Dea-Woo Park[2]

[1] Division of Computer Engineering, Sungkyul University, San 142-7,
Manan-gu, Anyang 8-dong, Anyang-city, Gyeonggi-do, Korea 430-742
Mopark777@hanmail.net
[2] Department of Computer Science, Soongsil University, Sangdo-dong 511,
Donggak-gu, Seoul, Korea 156-743
Prof1@hanmail.net

Abstract. The Global System for Mobile Communications (GSM) is the most popular standard for mobile phones in the world. GSM phones are used by over the billion people across more than 200 countries. In spite of the tremendous market growth, however, there are major security drawbacks in the GSM system. In this paper, we introduce the secure user authentication scheme to solve the problem: user authentication and location privacy. Also the proposed scheme provides partial anonymity, because of the usage of temporary identity and the new mechanism that the only authenticated VLR can use the MS's IMSI. Besides, we introduce the modified scheme to reduce user authentication procedure without changing of the architecture of the original GSM system.

1 Introduction

In the last few years, the analog cellular mobile telephones have supported reliable and ubiquitous communication services to people, and they also provide the idea of the mobility feature in the future PCS. However, in such open environments, all communications are transmitted as cleartext without any protection to prevent security threats, e.g., eavesdropping and illegal access. Therefore, to guarantee the security and satisfy the high quality requirement of more convenient and more various communication services, the so-called second generation digital cellular mobile telecommunications networks have already been developed and rapidly growing, such as, Global Systems for Mobile Telecommunications (GSM) and Digital European Cordless Telecommunications(DECT) in several European countries. The Global System for Mobile communications, GSM is widespread across the world and has always been the standard of the Pan-European digital cellular system. GSM is undoubtedly a major achievement in modern cellular telephony. GSM is so convenient in that anyone can use it to communicate with anyone else in almost any place at any time. There are a lot of subscribers across the world [1].

However, the GSM system has major worries about security weakness: the privacy of radio transmission and the authentication of the user [2][3]. Privacy e.g., confidentiality refers to the guarantee that the communication are not intercepted by an

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 883 – 892, 2006.

eavesdropper. Authentication is carried out to ensure that any unauthorized user cannot fraudulently obtain his/her required services from the home domains. In order to improve the security weakness in the GSM system, many researches proposed solutions. Harn and Lin's scheme [5] is to reduce the amount of information and eliminate the stored sensitive information in VLR for the GSM system. Lee et al. [6] presented the security mechanisms for the global architecture. For example, they proposed the confidentiality and key generation mechanism between one HLR and the other HLR. The architecture of the original GSM system was changed in many approaches [5][7][8]. Molva, Samfat and Tsudik [9] presented an efficient user authentication scheme with anonymity based on KryptoKnight [10]. Their scheme was based on the private key cryptosystem and focused on user authentication. Most of papers, however, do not provide user's anonymity at all. Our scheme is based on the private key cryptosystem to achieve the goal that uses conventional GSM system largely. We use the temporary identity e.g., TID in order to providing secure user authentication and location privacy. In this paper, TID plays an important role in providing the partial anonymity. We also introduce new scheme to reduce the TMSI allocation procedure without change the architecture of the original GSM system. We organize this paper as follows. In section 2, we describe privacy concept and problems on privacy. We also specify current researches on privacy of GMS system. In section 3, we present new user authentication providing enhanced location privacy and the partial anonymity in order to prevent the location privacy drawbacks of original network. We discuss and compare our protocols with the existed authentication approaches in many aspects of mutual authentication, anonymity, security, the reduction of bandwidth, and so on in section 4 and 5. In section 6, we conclude this paper.

2 User Authentication Protocol in the GSM

2.1 User Authentication Procedure

In the GSM system, the subscriber is initially registered in the HLR(Home Location Register) with a unique identity, IMSI, and obtains one secret key, Ki, from the AuC(Authentication Center) during the registration process. HLR is a database used for mobile information management. All permanent subscriber data are stored in this database. VLR is the database of the service area visited by an MS(Mobile Station). Two location databases play important roles in subscribers' registration and authentication [11]. The MS roams from one place to another and has access to the network in any place at any time.

The authentication process to updating MS's location with confidentiality is summarized as follows:

Step 1. The MS transmits the registration request(location update) to the new VLR(VLRn). The registration request includes the temporary mobile subscriber identity(TMSI) and LAI(Location Area Identity).

Step 2. Once the new VLR receives the TMSI, it sends a request to the old VLR(VLRo) asking for the authentication parameters for that MS.

Step 3. The old VLR sends MS's IMSI parameter to the new VLR after searching for IMSI corresponding to TMSI and LAI in its database.

Step 4. The new VLR forwards the IMSI to the HLR to asking for the MS's authentication.

Step 5. The HLR computes SRES and Kc by applying the MS's secret key Ki and a RAND number to the A3 and A8 algorithms, and then it sends the authentication triplet (RAND, SRES, and Kc) to the new VLR.

Step 6. The new VLR sends the RAND to the MS, and asks the MS to compute the SRES and sends it back.

Step 7. The MS computes the SRES and the Kc locally using that RAND number and the Ki through the A3 and A8 algorithms, then sends SRES back to the VLR and keeps Kc for later use.

Step 8. The new VLR once receives the SRES from the MS, compares it with the SRES provided from the HLR. If the two are equal, the MS passes the authentication process.

2.2 Security Drawbacks of User Authentication Protocol

Since the GSM system does not adopt ciphering mechanism between VLR and VLR/HLR, an eavesdropper can monitor the physical channel that connects to the HLR and eavesdrops MS's location updating information and information related security[4]. These drawbacks of GSM system enlarge the possibility of the privacy violation on users. Thus it is found that the authentication and location privacy protocol in GSM system have some drawbacks as follows[6][12]:

- When the VLR updates the location of MS, IMSI is exposed and delivered throughout the network without any protection. This is the big problem in user authentication protocol.
- Mutual authentication mechanism between MS and VLR does not provided. The GSM system only provides unilateral authentication for the MS. Using the challenge and response mechanism, the identity of a MS is verified. However, the identity of VLR cannot be authenticated. It is therefore possible for an intruder to pretend to be a legal network entity and thus to get the MS' credentials.
- The VLR must turn back to the HLR to make a request for another set of authentication parameters when the MS stays in the VLR for a long time and exhausts its set of authentication parameters for authentication. There is bandwidth consumption between the VLR and HLR.
- Every MS in the VLR has *n* copies of the authentication parameters. The parameters are stored in the VLR database, and then space overhead occurs.
- Authentication of MS is done in the VLR and this must be helped by the HLR of the MS for each communication.
- When a user roams to another VLR, the location is updated by sending IMSI to the new VLR while the old VLR is not accessible and no correct subscriber data is available. It is possible that an unauthenticated third party may eavesdrop on the IMSI and identify this mobile user.

3 Secure User Authentication Protocol

In this chapter, we introduce security-enhanced user authentication protocol to solve the problems that occur during TMSI allocation procedure in the GSM system. We name the proposed scheme as SUALPPA in the meaning that it provides the following major functions: Secure User Authentication, Location Privacy, and Partial Anonymity.

3.1 SUALPPA

The proposed scheme uses MS's temporary identity e.g., TID for partial anonymity. Here, "partial anonymity" has literally the meaning that guarantees partially user anonymity in new user authentication scheme. In this paper, that is to say, the old VLR provides MS's TID instead of MS's IMSI to the new VLR before success of the new VLR authentication by the HLR of the MS. The new VLR can acquire MS's IMSI only after authentication of the HLR. Thus user's anonymity is provided until the new VLR is authenticated by the HLR. To avoid the location tracking, we use the TID, which is mapped by one-to-one with the IMSI. So the TID must be unique as an additional parameter to authenticate user instead of the IMSI. The relation between the TID and the IMSI is kept secretly only by the HLR and the MS. But, the parameter TID itself is public information. And only the HLR can generate user's new TID. User can take together a new TID during the registration process that he/she obtains one secret key Ki and the IMSI.

The HLR gives the new VLR authorization to authenticate the MS. But, the new VLR processes authentication of a MS without knowing the secret key Ki of the MS. If the MS stays in the coverage of its new VLR for a long time, the VLR does not go back to the HLR to require another set of authentication parameters to identify the MS. The new VLR only uses the temporary key TKi of the HLR given with its generated RANDj for each call to compute the SRES and then identifies the MS, where RANDj is a random number generated by the new VLR in the subsequent calls. Only one RANDj is generated by the new VLR for each jth call no matter how long the MS stays in the coverage of the new VLR. This operation will be done only once in the first call when the MS visits at the new VLR.

In order to endow the new VLR with MS authorization, the HLR of the MS requires the legality of the new VLR. In this paper, we use a certification to check the legality of the new VLR. The HLR generates a certification of the VLR after finishing authentication of the new VLR. We notate a certification of the VLR as $Auth\_VLR_V$. Here, the certification $Auth\_VLR_V$ is different from the general certification in the public cryptosystem. The compositions of the VLR certification, $Auth\_VLR_V$ are T1, T2, Kvh, and $RAND_V$. (Kvh, X3) and (Kvh, $RAND_V$) pairs are inputted into the A3 algorithm and the two 32-bit results are combined to obtain the result of $Auth\_VLR_V$. Here, Kvh is a secret key shared between the new VLR and the HLR. And X3 is produced by computation T1 XOR T2 XOR $RAND_V$ and the notation XOR means the XOR operation.

Table 1. Notation

T1	Timestamp generated by the MS
T2	Timestamp generated by the new VLR
TID	Temporary identity of the MS
RAND1, RAND_V	Random numbers generated by the new VLR
RAND_H	Random number generated by the HLR

The security of the proposed scheme is based on the conventional architecture of the GSM authentication mechanism e.g. A3, A5 and A8 algorithms. Despite that the inputs of A5 algorithm are made up of 64-bits and 22-bits in the original GSM system, the output of A3 algorithm is designed to be 32-bits[12]. The length of TKi should be 64-bits in the proposed scheme. The method to generate a 64-bits TKi is to run two times A3 algorithm in the HLR and the MS. (Ki, RAND_H) and (Ki, T1) pairs are inputted into A3 and the two 32-bit results are combined to obtain 64-bits of TKi. The proposed scheme will achieve the following main design goals:

- Secure user authentication and location privacy
- Mutual authentication
- Secure distribution of a IMSI
- Partial anonymity

Also the proposed scheme has the following additional design goals.

- The new VLR authenticates a MS
- Reduction of the stored space in the VLR
- Reduction of bandwidth consumption between the VLR and the HLR

The procedures of SUALPPA are as follows.

Step 1. The MS sends TMSI, LAI, and a time-stamp T1 to the new VLR. T1 is to authenticate the new VLR and prevents it from replay attack.

Step 2. After receiving TMSI and LAI from the MS, the new VLR forwards TMSI and LAI to the old VLR in order to obtain the MS's TID.

Step 3. The old VLR sends the TID to the new VLR after searching for the TID corresponding to TMSI and LAI in its database. If there is no IMSI that is corresponded to the TID, then the session may be terminated. In the conventional process in GSM, the old VLR sends an IMSI instead of a TID. So, there is a problem that the network entities can easily obtain the sensitive information, IMSI.

Step 4. The new VLR generates RAND_V and a timestamp T2, which are used to authenticate the VLR itself to the HLR and computes Auth_VLR_V according to its generation method. And then the new VLR sends the TID along with its identification VLR_ID, T1, T2, RAND_V and Auth_VLR_V to the HLR. The HLR uses Auth_VLRv as an authentication parameter to authenticate the legality of the VLR.

Step 5. Once receiving the parameters, the HLR checks if the identity VLR_ID is a legal VLR. And then the HLR computes X3 by using T1, T2, and RAND_V transferred from the VLR. Since the HLR knows the secret key

Kvh shared between the VLR and the HLR corresponding to the VLR_ID, it can compute the value Auth_VLR$_V$' through A3 using (Kvh, X3) and (Kvh, X3) to authenticate the VLR. If the value Auth_VLR$_V$' and Auth_VLR$_V$ are same, the HLR believes that the new VLR is correct. And then the HLR computes the certificate of the new VLR to be sent to the MS, Auth-VLR$_H$ through A3 using Ki and T1. Also the HLR generates RAND$_H$ and computes TKi through A3 using (Ki, RAND$_H$) and (Ki, T1). After generating TKi, the HLR computes A5(IMSI, TKi) through A5 using the secret key Kvh and the IMSI related to the transmitted TID. The HLR finally sends the identity of the HLR e.g., HLR_ID, Auth-VLR$_H$, RAND$_H$, T1, and A5(IMSI, TKi) to the new VLR.

Step 6. After checking the HLR_ID, the new VLR extracts MS's IMSI and TKi using the secret key Kvh. By this processing, the VLR can know the IMSI and TKi, which is the temporary key to authenticate the MS. And then the VLR generates the random number RAND1. In the next call, the VLR should generate another random number. That is to say, as long as the MS stays in the coverage of the new VLR, the VLR does not need to go back to the HLR to require another set of authentication parameters. The VLR only generates a different RANDj for each jth call. The VLR sends T1, RAND1, RAND$_H$, and Auth-VLR$_H$ to the MS.

Step 7. Upon receiving the parameters, the MS first checks if T1 is the same as it was when last sent. If the result is valid, the MS computes Auth-VLR$_H$' through A3 using Ki and T1 and then compares Auth-VLR$_H$' with the received Auth-VLR$_H$. If two certification values are the same, the MS believes the new VLR and generates TKi according to the generation method. The MS continues through A5 using TKi and RAND1 as inputs to generate the SRESm, which is then sent back to the new VLR.

Step 8. The VLR compares SRESm' with SRESm. The VLR computes SRESm' in advance before being transmitted SRESm from the MS. If they are the same, the authentication is finished.

3.2 Modified SUALPPA

In this section, we modify the proposed scheme to reduce the numbers of the authentication transaction. We notate the modified scheme MSUALPPA in table 2. The basic concept of MSUALPPA is the same as one of the first proposed scheme. However, there are two different points in the procedure. One difference is that the TID is added to the MS's transfer parameter in the first step. Thus the MS sends the parameter T1, TMSI, LAI, and TID to the new VLR in the first step of the second scheme. The other is that the 2nd and 4th steps in the first proposed protocol are simultaneously performed of the second one. So, the 3rd and 5th steps are automatically performed after being completed them respectively. In other words, the new VLR sends immediately the TID that is sent from the MS to the new VLR, to the HLR after completing the 1st step without waiting for the sending of the TID from the old VLR in the 3rd step because the new VLR has already the TID sent from the MS in the 1st step.

Table 2. Procedures of the proposed schemes

Step	SUALPPA		Step	MSUALPPA
1	MS→VLRn:TMSI,LAI,T1		1	MS→VLRn:TID,TMSI,LAI,T1
2	VLRn→VLRo:TMSI, LAI		2	VLRn→VLRo : TMSI, LAI
				VLRn→HLR: TID, VLR_ID, T1, T2, RAND$_V$, Auth_VLR$_V$
3	VLRo→VLRn: TID		3	VLRo→VLRn: TID
		=>		HLR→VLRn: A5(IMSI, TKi), Auth_VLR$_H$, HLR_ID, T1, RAND$_H$
4	VLRn→HLR:T1,VLR_ID, TID,RAND$_V$, Auth_VLR$_V$, T2		4	VLRn→MS: RAND1, RAND$_H$, T1, Auth_VLR$_H$
5	HLR→VLRn: Auth_VLR$_H$, HLR_ID, A5(IMSI, TKi), RAND$_H$, T1		5	MS→VLRn:SRESm
6	VLRn→MS: RAND1, T1, RAND$_H$, Auth_VLR$_H$			
7	MS→VLRn:SRESm			

4 Cryptanalysis

Owing to the fact that we adopt the architecture of the conventional authentication scheme in GSM, the security of the new scheme, which is the same as that of the existing authentication scheme in GSM, is based on algorithms A3, A5 and A8. In order to authenticate the legality of the new VLR and the MS, we add a time-stamp T1 and T2 to the user authentication protocol. The time-stamp T1 and T2 enhance the security of the proposed scheme against replay attack. Although an attacker can intercept T1, T2, RAND$_V$ and Auth-VLR$_V$ and then forge the real VLR, the replay still cannot succeed because T1, T2, and RAND$_V$ are incorrect. The MS can also check if the T1 is the same as it was when sent the last time even if the fake VLR replays T1 and Auth-VLR$_V$.

The MS verifies the new VLR by the Auth-VLR$_H$ transmitted from the HLR. Nobody can forge it to fool others, since the secret key Ki is known only to the MS and the HLR. Without the knowledge of Ki, Auth-VLR$_H$ cann't be computed by anyone. Therefore, the security of the proposed protocol is based on Ki. Also, the Auth_VLR$_V$ is made stronger than the other certification schemes of the new VLR. For authenticating the MS, the new VLR only generates a different RANDj to compute SRESm for every jth call. The security here is based on the HLR giving the new VLR the authorization to authenticate the MS. Nobody can suppose the value IMSI with the TID, since only the HLR knows the relation between the TID and the IMSI. Besides, there is no the exposure of the IMSI in wired channel, since the only authenticated VLR can use the IMSI, which is transferred in encryption mode to the VLR.

5 Discussions

In this section, we shall demonstrate that our proposed schemes can achieve our requirements.

- Mutual authentication between the MS and the VLR: The HLR uses Auth_VLR$_V$ to identify the VLR and Auth_VLR$_H$ to transmit to the MS the fact that the VLR has been verified by the HLR of the MS. By authenticating Auth-VLR$_H$ transmitted from the HLR, the MS can ensure that it is communicating with a legitimate VLR.
- Reduction of bandwidth consumption: The HLR gives the VLR temporary secret key TKi to authenticate the MS. As long as the MS stays in the coverage area of the new VLR, the VLR can use the TKi to authenticate the MS for each call. Since the new VLR does not go back to the HLR to require another set of authentication parameters, the signaling load is reduced between the VLR and the HLR.
- Reduction in the storage of the VLR database: The VLR only stores one copy of authentication parameter instead of n copies (RAND, SRES, Kc).
- The proposed schemes do not add any computations to it, nor is there any change in the original architecture of the GSM system in order not to lose simplicity and efficiency advantages of the GSM system, which is widespread in the world. The security of the proposed schemes is still based on algorithms A3, A5 and A8.
- Authentication of the MS by the new VLR: Authentication of a mobile user is done by the new VLR instead of the HLR except the first call for TMSI allocation, even if the VLR doesn't know the subscriber's secret Ki.
- The only VLR that is authenticated by the HLR of the MS can use MS's IMSI: The conventional papers and the GSM system assume that the VLR is a legal entity. In this paper, the HLR believes the new VLR according to the verification result after authenticating it without any assumption. Thus the proposed protocols are more secure than the others.
- Partial anonymity: Our approaches provide the mobile user with partial anonymity by using the TID between the new VLR and the old VLR. The conventional approaches don't provide partial anonymity at all. The procedure to provide partial anonymity brings the effect to reducing encryption process, since the procedures e.g., from the 1st to the 4th step that offer partial anonymity don't use the encryption.
- Secure location privacy: This is the most important goal in this paper. Our approaches used the TID instead of the IMSI between the new VLR and the old VLR. It is possible for the new VLR and any entities to acquire the IMSI only after the HLR authenticates them. When the HLR transfers the IMSI to the new VLR, the IMSI is sent in the encrypted mode by using the shared secret key between the HLR and the VLR. Thus the value IMSI isn't exposed the unauthenticated entities.

The conventional protocols not only do not meet all our requirements, but they also change the architecture of the GSM authentication protocol. Our schemes keep the advantage of not changing the architecture of the GSM system. Lee et al. [6] proposed the protocol that does not change its architecture. However, it doesn't provide mutual authentication between the MS and the VLR. The GSM system doesn't also support mutual authentication. Since the VLR doesn't ask the HLR for another set of authentication parameters in Lee et al.'s and our protocols, the bandwidth consumption is less than that of the original GSM protocol. In addition, the VLR requires storage of n copies of the authentication parameters in the original GSM protocol. In Lee et al.'s and our protocols, the VLR only requires storage of one copy of the authentication parameters instead of n copies in its database.

Partial anonymity and the point that only verified VLR can use the IMSI are provided only in our paper. Also the subject of the IMSI assignment is done by the HLR instead of the old VLR. The procedure that the HLR sends the IMSI to the new VLR doesn't need the additional computations to search and compute the IMSI, since the HLR already has the IMSI in its database and it uses the value IMSI in the original GSM system to check the correct MS. Thus the HLR has the responsibility on distribution of the important information. In the original GSM system, there are at least six messages transmitted during location update for the MS. In our second proposed scheme, the number of data flows is reduced and it is shown in the table 2. The followings are the meanings of the abbreviated words in table 3. MA: Mutual authentication between the MS and the VLR, RBC: Reduction of bandwidth consumption, RSV: Reduction of storage in the VLR, EVV: Encryption between the old VLR and the new VLR, PA: Partial anonymity, AI: Assignment of IMSI, UAV: The use of IMSI after authentication of the VLR, RF: Reduction of data flow in protocol, CAG: Change the architecture of the GSM system.

Table 3. Comparison among GSM authentication protocols

	GSM	Ours1	Ours2	[6]	[9]	[12]
MA	N	Y	Y	N	Y	N
RBC	N	N	N	Y	N	Y
RSV	N	Y	Y	Y	N	Y
EVV	N	N	N	Y	Y	Y
PA	N	Y	Y	N	N	N
AI	VLR	HLR	HLR	VLR	VLR	VLR
UAV	N	Y	Y	N	N	N
RF	-	N	Y	Y	Y	N
CAG	-	N	N	N	N	N

6 Conclusions

We have proposed the new user authentication schemes that can satisfy our requirements in order to overcome drawbacks of user authentication and location privacy in the original GSM system. Our schemes have some advantages addressed in section 4 and 5 like other schemes. Also, our schemes provide additionally the partial anonymity and the secure use of an IMSI by the only VLR that is verified by the HLR. Besides, our schemes offer the reduction of the total data flows by using a TID without changing the architecture of the original GSM system.

References

1. Jorg Eberspacher, Hans-Jorg Vogel and Christian Bettstetter: GSM, switching, Services and Protocols (2nd edition), WILEY(2001)
2. D. Broron: Techniques for privacy and authentication in personal communication systems, IEEE Personal communications (8.1995) 6–10
3. J.E. Willas: Privacy and authentication needs of PCS, IEEE personal communications (8.1995) 11-15
4. S.P. Shieh, C.T. Lin, J.T. Hsueh: Secure communication in Global Systems for Mobile Telecommunications, Proc. 1st Workshop on Mobile Computing, ROC. (1995) 136–142
5. HARN, L. and LIN, H.Y: Modification to enhance the security of the GSM protocol, Proceedings of the 5th National Conference on Information security, Taipei, Taiwan, May. (1995) 416–420
6. Lee C.C., Hwang M.S., Yang, W.P.: Extension of authentication protocol for GSM. IEE Proceedings. Communications, Vol. 150, No.2, (2003) 91-95
7. Al-Tawil, K., Akrami, A., and Youssef, H.: A new authentication protocol for GSM networks, Proceedings of IEEE 23rd Annual Conference on Local Computer Networks(LCN'98), (1998) 21–30
8. Stach, J.F., Park, E.K., and Makki, K.: Performance of an enhanced GSM protocol supporting non-repudiayion of service, Comput. Commun., 675-680 (1999)
9. Molva, R., Samfat, D., Tsudik, G.: Authentication of mobile users, Network, IEEE Volume 8, Issue 2, (1994) 26–34
10. R. Molva, G. Tsudik, E. V. Herreweghen, and S. Zatti. KryptoKnight: Authentication and key distribution system. In Proceedings on 1992 European Symposium on Research in Computer Security, (1992) 155–174
11. K. Chae and M. Yung (Eds.): WISA 2003, LNCS 2908, pp. 162.173, 2004. Springer-Verlag Berlin Heidelberg 2004, A Location Privacy Protection Mechanism for Smart Space
12. Chii-Hwa Lee, Min-Shiang Hwang, and Wei-Pang Yang: Enhanced privacy and authentication for the global system for mobile communications, Vol. 5, No.4, Wireless Networks 5 (1999) 231–243

Design of Mobile Video Player Based on the WIPI Platform*

Hye-Min Noh[1], Sa-Kyun Jeong[1], Cheol-Jung Yoo[1], Ok-Bae Chang[1],
Eun-Mi Kim[2], and Jong-Ryeol Choi[3]

[1] Dept. of Computer Science, Chonbuk National University, 664-14 1ga, Duckjin-Dong,
Duckjin-Gu, Jeonju, Jeonbuk, South Korea
{hmino, umin, cjyoo, okjang}@chonbuk.ac.kr
[2] Division of Computer & Game, Howon University, 727, Wolha-Ri, Impi-Myeon,
Kunsan, Jeonbuk, South Korea
ekim@sunny.howon.ac.kr
[3] College of Culture and Tourism, Jeonju University, 1200 3ga, Hyoja-Dong,
Wansan-Gu, Jeonju, Jeonbuk, South Korea
god22c@dreamwiz.com

Abstract. The mobile video player that supports software decoders has a different structure from the video player that plays videos based on a hardware decoder. Therefore, this paper elucidates the design a mobile video player consisting of a network manager, displayer, event processor, data structure processor, and controller based on research regarding hardware restriction, a software development platform, and streaming implementation method, all of which were applied before implementing the video player at the WIPI platform. The mobile video player implemented based on the designs investigated during this research can be used as a tool to test function and the possibility of normal operation of a software decoder that targets an encoded visual, and also can be used as a tool to develop an improved software decoder. The design of a mobile video player in this paper follows from basic research on the construction of a mobile video total system based on the WIPI platform in the future.

1 Introduction

Platforms used in mobile phones exist in diverse forms, depending on the communication companies that produced them. In the case of Korea, GVM and SK-VM of SK Telecom, BREW and MAP of KTF, and KVM of LG Telecom correspond to such platform[1]. The problem with diverse platforms is that overlapped development of individual video player development and multimedia methods of the next generation for mobile phones waste development expenses. Therefore, an effort to standardize platforms has been made, and from this research the WIPI platform was developed.

* This work was supported by grant No. R01-2004-000-10730-0 from the Basic Research Program of the Korea Science & Engineering Foundation.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 893 – 898, 2006.

The development of a mobile video player in a standard platform can reduce costs compared to the former method of developing each platform separately. Also, mobile a video player based on the standard platform has the strength of playing videos without requiring an additional hardware chip because it can combine the video decoder with the software method. Therefore, there is need for a video player in a different form compared to the mobile video player combined with the former decoder based on a hardware chip, and the type of decoders must be distinguished according to mobile phones along with the existence of a software decoder. Additionally, there is need for the design of an additional function other than the former mobile player based on a hardware decoder such as decoder manager.

This paper analyzes the properties and structure of a mobile video player to support a software based decoder as the first level of implementation required to investigate the design and implementation of the WIPI platform[2] based mobile video player. Also, this paper outlines the design of a mobile video player appropriate for such conditions.

2 Considerations for Designing Mobile Video Player

A hardware executing application, platform that executes the software, network environment, and oversees other basic content must be considered in all its complexity when designing and implementing a video player.

The video player to be designed in this paper is the application executed in mobile phones including the WIPI platform among mobile phones[2]. Such a mobile phone has inferior resources compared to normal the PC environment that considers hardware resources. Therefore, the following are the details to be considered in implementing video player executed in mobile phones.

First, the restriction of hardware; there is restriction in hardware function and display when operating the UI(User Interface) of a player[3]. Mobile phones have much a significantly more restricted display size than the size of display expressed in a normal PC environment. The video player designed in this paper is the application supported by normal mobile phones selected by the WIPI platform other than devices including operating systems such as WINDOWS CE selected by PDA-type mobile devices.

Thus, the format of vide input and output is based on the video frame format of S-QCIF(Sub-Quarter Common Intermediate Format) or QCIF(Quarter Common Intermediate Format) among CIF formats for the process of video input and output. The size of a LCD display of mobile phones includes mobile phones that have the size of a LCD display that can process video of such frame format. The video frame format is related to not only the display resolution, but also to power consumption used for decoder. Also, the hardware function defines CPU function of mobile phones normally used, and CPU currently used widely in mobile phones is ARM7 or ARM9 versions. Considering that ARM9 has many aspects of video decoding through the decoder and execution of the player, ARM7 has comparably better functions. Thus, this research designed the platform based on functions of ARM9.

Secondly, a platform that executes software should be considered. This paper aims to design and implement a video player in the WIPI platform. In other words, all functions of the video player are designed with a focus on implementation based on Jlet in the WIPI platform.

Thirdly, streaming needs to be considered since there is no allotment of large-scale memory unit as a data space for video playing due to hardware restrictions. For streaming, the research conducted for this paper designed parts of a network processor, and designed not only the network for streaming, but also file management part to take care of video data saved on the local memory. The file management part also plays an important role in player application since the data to be processed increases per second for streaming.

In this way, there are many details to consider compared to a normal environment with abundant resources for many parts to design a mobile video player. Such considerations must not be investigated one by one, but rather through a holistic approach. For example, the power consumption must be considered by considering not only the size of the LCD display, but also the fact that it involves mobile phones when selecting the video frame format of mobiles, and to process streaming data, the network speed and size of device memory must be considered and selected for the amount of momentary data process.

3 Structure and Design of Mobile Video Player

The structure of a mobile video player is composed of a network manager that executes decoder search and a download function that can process video data and transmit it to the decoder by receiving video data from a streaming server, displayer to highlight the decoded video data on display, an event processor to process all events that occur in the player, data structure processor to pass the data developed in decoder to the displayer, and controller to process all these together.

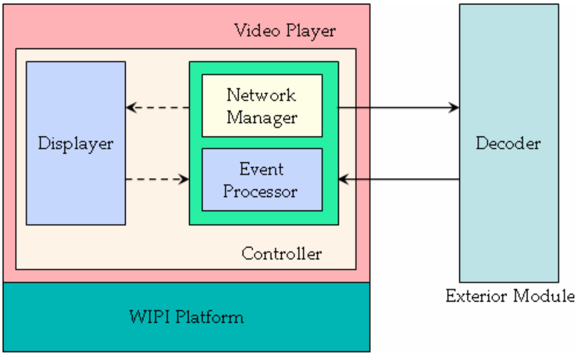


Fig. 1. Video Player Structure

3.1 Network Manager

The network manager is divided into a RTP/RTCP[4][5] interface network controller to process streaming data, a file manager that processes data from streaming, and a decoder manager to manage the downloaded decoder and download the appropriate decoder onto server.

The network controller requests streaming data from the server and processes the data from the server. Also, it synchronizes, controls in sequence, and processes errors for video data. Data transmission are transmitted of UDP(User Datagram Protocol) through RTP(Real-time transport Protocol) module, and the control information transmits TCP(Transmission Control Protocol through RTCP(RTP Control Protocol) module.

The file manager functions to prevent I/O bottleneck situation by appropriately processing data transmitted from mobile phones after streaming, and manages a buffer according to the capacity of memory unit of mobile phones.

The decoder manager identifies an encoding format of video input that executed to process the decoder, which is a necessary factor for video play. This device conducts a search to determine whether or not the decoder is installed in mobile phones for video play according to the encoding format, or functions to download the decoder by connecting it to the decoder server if not installed. Such functions are executed through TCP transmission.

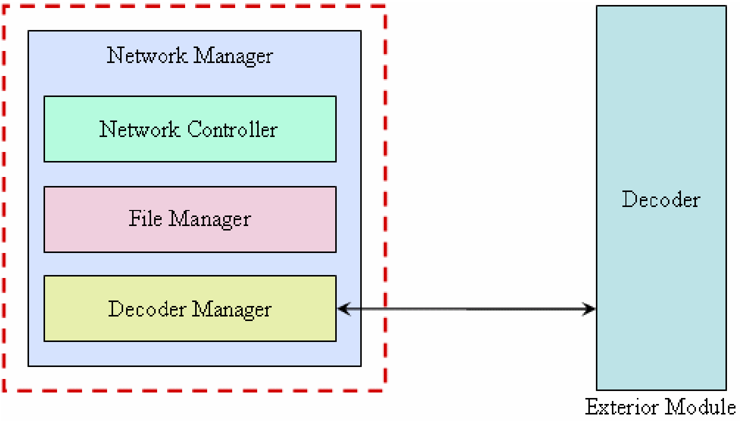


Fig. 2. Overall Module of Network Manager

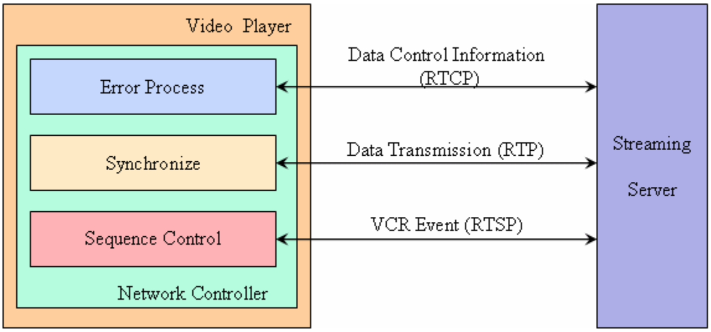


Fig. 3. Detailed Module of Network Controller

3.2 Displayer

The displayer functions to output video display decoded data on mobile phones using the WIPI platform card, which is output after delivering a transmitted unit of mobile data NAL(Network Abstract Layer) from mobile phones after streaming through the network. Also, the displayer has skin for video players.

3.3 Event Processor

The event processor functions to deliver VCR events(play, pause, stop, jump) made through a device key developed in players through RTSP module to servers[6], and plays a role in processing all events that develop in the video player.

3.4 Data Structure Processor

The data structure processor defines standard structure of data transmitted to the displayer as module to process data delivered to the displayer from the decoder, and functions to manufacture data format output from the decoder. This functions to buffer communication between decoder and displayer as an inner module.

3.5 Controller

The controller functions to take care of errors that can develop in the application and life cycle processes of video player application. In other words, the controller controls each function of the video player and processes overall application.

The decoder function is excluded from the mobile video player designed in this paper. The decoder forms a component format being downloaded independently. This means that the decoder does not include a chip in mobile phones in regards to hardware, or implementation in regards to software. If the decoder is implemented in such software format, diverse videos can be played that follow the encoding standard through the suggested video player.

4 Conclusion and Further Studies

The video players implemented in recent mobile phones are mainly video player formats developed according to each platform through a hardware decoder selected by each communication company. However, mobile phones that include the WIPI platform have been gradually and continuously launched as the WIPI platform has been confirmed as the mobile standard platform of the next generation. Such changes in the environmental conditions mean that each communication company is moving away from the environment of individual development. Moreover, the suggested mobile video player has the strength of adopting easily in diverse kinds of mobile phones since it does not depend on a hardware chip to decode the encoded video.

This paper suggests details that must be considered before implementing the mobile video player that has such strengths, and designs a mobile video player executed in the WIPI platform based on such considerations.

Based on the results of this research, solutions to problems encountered while implementing the WIPI platform provided in mobile phones based on actual ARM9 were identified, and an improvement in the technology for implementation in the future was achieved. The representative problem involves the weakness of not being able to elicit the hardware function 100% since the WIPI platform graphic package is a standardized package for all mobile phones. Therefore, there is need to research the coding format to elicit maximal graphic functions by analyzing the compilation process when mobile video controller application is downloaded after being compiled according to the hardware of each device, and to implement a player accordingly.

References

1. Si-woo Byun and Sook-eun Byun, “A study on WIPI Platform for Efficient Mobile Business,” Korea Information Science Society, Vol. 4, No. 2, pp. 79-93, 2003.
2. Mobile Standard Platform WIPI 2.0.1, Specification No. KWISFS.K-05-003, Sep. 2004.
3. Jae-Wook Yeou, Jae-Il Jung, Yong-Kyung Shin, and Sang-Wook Kim, “The MPEG-4 Video Player for PDA,” Korea Information Science Society, Vol. 29, No. 2, pp. 145-147, Oct. 2002.
4. “RTP:A Transport Protocol for Real-Time Applications,” Request For Comments(RFC) 1889, Internet Engineering Task Force, Jan. 1996.
5. Real Time Streaming Protocol, Request For Comments(RFC) 2326, Internet Engineering Task Force, Nov. 1998.
6. Hyung-Kook Jun and Pyeong-Soo Mah, “Design and Implementation of MPEG-4 Media Player Supporting QoS and Retransmission,” Korea Information Science Society, Vol. 29, No. 2, pp. 343-345, Oct. 2003.

Discovering Patterns Based on Fuzzy Logic Theory

Bobby D. Gerardo[1], Jaewan Lee[1], and Su-Chong Joo[2,*]

[1] School of Electronic and Information Engineering, Kunsan National University
68 Miryong-dong, Kunsan, Chonbuk 573-701, South Korea
{bgerardo, jwlee}@kunsan.ac.kr
[2] School of Electrical Electronic and Information Engineering, Wonkwang University
344-2 Shinyong-dong, Iksan, Chonbuk 570-749, South Korea
scjoo@wonkwang.ac.kr

Abstract. This study investigates the formulation of fuzzy logic as integrated component of the proposed model in data mining in order to classify the dataset prior to the implementation of data mining tools such summarization, association rule discovery, and prediction. The novel contribution of this paper is the fuzzification of the dataset prior to pattern discovery. The model is compared to the classical clustering, regression model, and neural network using the Internet usage database available at the UCI Knowledge Discovery on Databases (KDD) archive. Our test is anchored on parameters like relevant measure, processing performance, discovered rules or patterns and practical use of the findings. The proposed model indicates adequate performance in clustering, higher clustering accuracy and efficient pattern discovery compared with the other models.

1 Introduction

The current trend shows that intelligent system uses fuzzy logic and neural network theories to minimize uncertainty of data and in addition, the latter could provide fair learning performance by modeling human neural system mathematically [1].

Fuzzy logic had been claimed as one of the better techniques in connection with the human reasoning and decision making purposes. The fuzzy theory was coined by Zadeh in the paper fuzzy sets [2]. This fuzzy logic is a relatively young discipline, both serving as a foundation for the fuzzy logic in a broad sense and of independent logical interest, since it turns out that strictly logical investigation of this kind of logical calculi can go rather far. The use of fuzzy theory [3] could extend to business and finance, traffic control, automobile speed control, and earthquake detection.

In the other perspectives, varieties of data mining tools had been developed to address major applications in academic, business or industrial purposes. Some examples of these tools are used for concept description, discovering patterns, classification, prediction, and cluster analysis. Some constraints that most researchers observed in the data mining tasks are calculation speed, practical value of the methods

[*] This paper was supported by Wonkwang University in 2005.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 899 – 908, 2006.

used, reliability of the approach for computation, heterogeneity of database, and large amount of data to compute.

Despite of the efficiency of some neural network algorithms [1], [4], [5] in classification or clustering, it is noted that the performance or computation time is sometimes unbearable. This study will introduce the selected intelligent system theories and propose a modified model using one of the theories as integrated component of the data mining system. We will investigate the formulation of fuzzy logic as integrated component of the proposed model in order to classify the original dataset prior to implementation of other data mining tools such summarization, association rule discovery, and prediction.

The novel contribution of this paper is the fuzzification of the dataset prior to pattern discovery, this is presented in section 3. The model will be compared to the classical clustering, regression and neural network using the database available at the UCI KDD archive [6]. Our test is anchored on parameters like relevant measure, processing performance, discovered rules and practical use of the findings.

2 Related Studies

One of the essential processes in data mining is the association rule discovery rendered on from simple to complex database repositories in the distributed system. Association rule mining tasks are finding frequent patterns, associations, or causal structures among sets of items in transactional databases, relational databases, and other information repositories. Data mining uses various data analysis tools such as from simple to advanced mathematical algorithms in order to discover patterns and relationships in databases.

Cluster analysis is used for data analysis in solving classification problems. The goal of cluster analysis is categorization of objects so that the degree of correlation is strong between members of the same cluster and weak between members of different clusters. Such classification may help formulate hypotheses concerning the origin of sample, describe a sample, predict the future behavior of population types, and optimize the functional processes on classes within the population [7], [8].

The k-means algorithm is one of a group of algorithms called partitioning methods. This algorithm is primarily used for clustering tasks. Its means are used as the new cluster points and each object is reassigned to the cluster that it is most similar [9].

2.1 Intelligent System Theories

The most prominent theories in intelligent systems are fuzzy logic, neural network and regression theories, which will be discussed in the subsequent sections.

The Fuzzy Theory. Fuzzy systems are an alternative to traditional notions of set membership and logic that has its origins in ancient Greek philosophy, and applications at the leading edge of Artificial Intelligence. Yet, despite its long-standing origins, it is a relatively new field, and as such leaves much room for development [10]. Fuzzy theory had been utilized in various fields like in dynamic control environment, traffic accident prediction, information retrieval system, navigation systems for cars, automatic operations of trains and water level controller.

One significant application is in expert system where the theory is used for decision support, financial plan, diagnostics systems and information retrieval. In the study of Saint-Paul et al. [11] about general purpose database summarization, they used fuzzy set-based methods to construct robust summaries from datasets, using linguistics variables. Their process yields summary hierarchy which provides views on the data at different levels of granularity through perfectly understandable high level descriptors. In the study of Lee et. al. [1], the neural network, fuzzy, quantification and regression models were compared for accident frequency prediction. They noted that fuzzy and neural network models were superior to the other models mentioned.

The study on expert system using fuzzy logic shows improved performance based on the network traffic detection [3]. Their system has a knowledge base that stores rules used by the fuzzy inference engine to get a new fact from them.

Natural language abounds with vague and imprecise concepts [10], such as "Clark is heavy," or "It is very cold today." Such statements are difficult to translate into more precise language without losing some of their semantic value .The main notion about fuzzy systems is that truth values (in fuzzy logic) or membership values (in fuzzy sets) are indicated by a value on the range [0.0, 1.0], with 0.0 representing absolute falseness and 1.0 representing absolute truth. For example, let us consider the statement: "Barry is old." If Barry's age was 72, we might assign the statement the truth value of 0.76. The statement could be translated into set terminology as follows: "Barry is a member of the set of old people." The probabilistic approach yields the natural-language statement, "there is a 76% chance that Barry is old," while the fuzzy terminology corresponds to "Barry's degree of membership within the set of old people is 0.76". Brule [10] defined the membership function for fuzzy set as follows:

Definition. Let X be some set of objects, with elements noted as x. Thus, X={x}. A fuzzy set A in X is characterized by a membership function mA(x) which maps each point in X onto the real interval [0.0, 1.0]. As mA(x) approaches 1.0, the "grade of membership" of x in A increases.

A study in information retrieval [4] uses fuzzy theory to cluster information. Its method is partitioning a given set of documents into groups using a measure of similarity which is defined on every pairs of documents. Similarity between documents in the same group should be large, while it should be small for documents in different groups. In a related study, similar technique was used in bioinformatics and medical science research. For instance, Dembele and Kastner [12] used the Fuzzy C means in partitioning of data. Other methods like K-means or self-organizing maps only assign each sample to a single cluster. In addition, these methods do not provide information about the influence of a sample for the overall shape of the clusters.

In our approach we integrate the fuzzy logic to cluster the dataset prior to mining so we can determine the clustering of sample data, which will indicate acceptable criterion values for its membership to a cluster. Furthermore, each group of data would likely reveal association and denote influence of a sample to a given cluster.

The Neural Network. There are many innovative researches in the application of NN to information retrieval as cited in [4]. Their particular study on information retrieval deals with the use of neural network to handle vagueness and uncertainty in data. On the other hand, the study on traffic accident predictions [1] also used NN to provide fair learning performance on the model that they proposed.

In general, the NN approach to clustering tends to represent each cluster as a prototype. The prototype of the cluster does not necessarily have to correspond to particular data example or object. The attributes of an object assigned to a cluster can be predicted from the attributes of the cluster prototype. Self-Organizing Map is one of the most popular neural network models. It belongs to the category of competitive learning networks. As cited by Hollmen [5], the Self-Organizing Map is based on unsupervised learning, which means that no human intervention is needed during the learning and little is needed to be known about the characteristics of the input data.

Another NN model is the Multilayer Perceptron (MLP). This class of networks consists of multiple layers of computational units, usually interconnected in a feed-forward way. Each neuron in one layer has direct connections to the neurons of the subsequent layer. In many applications the units of these networks apply a sigmoid function as an activation function. Multi-layer networks use a variety of learning techniques, the most popular being the back-propagation.

Regression Theory. Another theory used for intelligent system is the regression analysis. Its primary application is predicting continuous values rather than the other data types. Linear regression is the simplest form of regression analysis. The simplest form of a regression model contains a dependent variable also called outcome variable and a single independent variable also called factor.

2.2 Data Mining Processes and the Association Rule Algorithm

There are varieties of data mining algorithms that have been recently developed to facilitate the processing and interpretation of large databases. One example is the association rule algorithm, which discovers patterns in databases. The use of such algorithm is for discovering association rules. All rules that meet the confidence threshold are reported as discoveries of the algorithm.

3 Architecture of the Proposed Data Mining System

Based on the literatures that had been reviewed, we developed the proposed architecture for the data mining system as shown in Figures 1 and 2, respectively. The preprocessing stage as shown by funnel symbol in the figure includes data cleaning that performs data extraction, transformation, and loading.

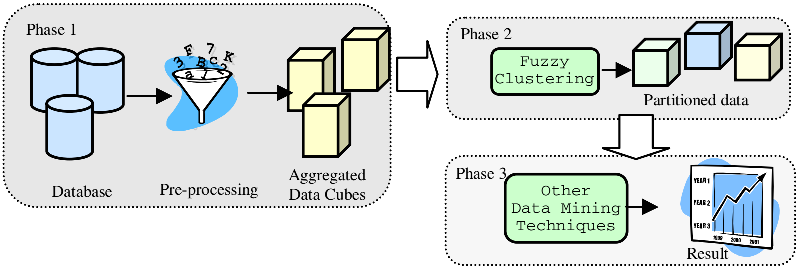


Fig. 1. The Proposed 3-Stage Data Mining Process

This stage also detects missing or outlier data and applies necessary filtering. In addition, another filtering process to transform inappropriate data is rendered on this stage just incase the data type being preprocessed is not suitable to the algorithm that is used. This will result to an aggregated data cubes as illustrated in the same figure. The Phase 2 in Figure 1 shows the implementation of the classification while Phase 3 is the implementation of data mining process to generate association rules.

The refinement of the processes is presented in Figure 2. In Phase 2 the fuzzy processes are indicated by the bubbles as shown in the fuzzy clustering box.

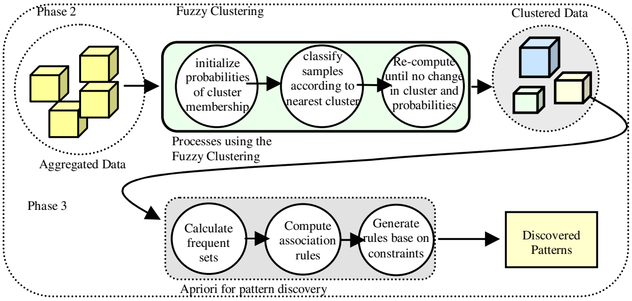

Phase 2
Fuzzy Clustering
Clustered Data
initialize probabilities of cluster membership
classify samples according to nearest cluster
Re-compute until no change in cluster and probabilities
Aggregated Data
Processes using the Fuzzy Clustering
Phase 3
Calculate frequent sets
Compute association rules
Generate rules base on constraints
Discovered Patterns
Apriori for pattern discovery

Fig. 2. Process view of the fuzzy clustering and the pattern discovery

Phase 3 is the final stage in which the association rule algorithm will be employed. The refinement of the process shows that the frequent itemsets will be calculated and compute for the association rules using algorithms for support and confidence. The output is given by the last rectangle showing the discovered rules.

4 The Fuzzy Algorithm and Performance Analysis

4.1 Fuzzy Algorithm

The Concept of Fuzzy K-Means. The Fuzzy K-Means Clustering is an extension of the K-means clustering method. This particular clustering uses the method called the Fuzzy K-means. If we assume that all the data points are not known, then we can use the Sequential K-Means by gradually obtaining such data points over a period of time. The goal is to find such partitions of a set of n samples which maximize the criterion function like the distance [9]. Fuzzy clustering can be applied as an unsupervised learning strategy in order to classify data.

The concept of Fuzzy comes from the Fuzzy logic, is an extension of Boolean logic dealing with the concept of partial truth. Whereas classical logic holds that everything can be expressed in binary terms (0 or 1, yes or no), fuzzy logic replaces Boolean truth values with degrees of truth [11]. Degrees of truth are often confused with probabilities. However, they are conceptually distinct; fuzzy truth represents membership in vaguely defined sets not likelihood of some event or condition.

Fuzzy K-Means Algorithm. For each iteration of the classical k-means procedure, we assumed that each feature vector belongs to exactly one cluster. We can relax this condition and assume that each sample Xi has some graded or "fuzzy" membership in a cluster μj. The probabilities of cluster membership for each point are normalized as:

$$\sum_{i=1}^{c} P(\omega_i|x_j) = 1 \text{, where } j = 1 \ldots \text{n} \qquad (1)$$

Each μ_i is then re-calculated as:

$$\mu_j = (\sum_{j=1}^{n} P(\omega_i|x_j)^b x_j)/(\sum_{j=1}^{n} P(\omega_i|x_j)^b) \qquad (2)$$

and each $P(\omega_i|x_j)$ is then re-calculated as:

$$P(\omega_i|x_j) = (1/d_{ij})^{1/(b-1)} / (\sum_{r=1}^{c} (1/d_{rj})^{1/(b-1)}) \qquad (3)$$

where the distance, $d_{ij} = \|x_j - \mu_i\|^2$ is calculated between each sample X_i and cluster μ_j. The pseudo code of the fuzzy clustering is presented below:

```
begin
   initialize n; $\mu_1$ ... $\mu_c$; $P(\omega_i \mid x_j)$, where i = 1,..., c and j = 1, ..., n
   normalize probabilities of cluster memberships
   do
      classify n samples according to nearest $\mu_i$
      recompute $\mu_i$
      recompute $P(\omega_i \mid x_j)$
   until no change in $\mu_i$ & $P(\omega_i \mid x_j)$
   return $\mu_1$ ... $\mu_c$
end
```

This shows that the graded membership, which fuzzy K-means offers, improves the convergence of the algorithm [13]. This implies that the convergence is better than the classical K-means algorithm.

4.2 Accuracy Measure

To optimize the process, we will use the fuzzy clustering agent prior to the association mining. This process is shown by the fuzzifier box in Figure 3. This will enable to cluster highly correlated attributes in groups which will result to an aggregated data as indicated in the same figure. The fuzzification is a process using membership function in order to classify the data placed in the input data box into clusters (see Figure 3).

The membership accuracy (MA) according to predefined clusters can be measured using the equation provided in the literature [3]. The MA is equal to correctly classified patterns (CCP) divided by total patterns (TP), where 1.0 being the highest MA and 0 is the lowest. The formula for membership accuracy is shown in Equation 4.

$$MA = \frac{CCP}{TP} \tag{4}$$

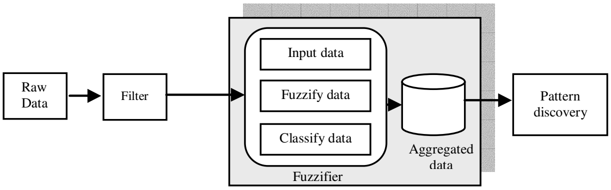


Fig. 3. Dataflow diagram of the proposed Fuzzy mining model

4.3 Performance Analysis

We used our previously generated dataset to test the proposed model. It contains 30 attributes and 500 instances of transactional dataset synthetically and randomly generated. A total of 4 clusters had been identified and it is presented in Table 1. In summary, cluster 1 has a total of 15 cases, cluster 2 has 100 cases, cluster 3 has 109 and cluster 4 has 276 cases. For classical clustering, cluster 1 has 15 cases, cluster 2 has 99, cluster 3 has 108 cases and cluster 4 has 271 cases.

Table 1. Comparison of fuzzy and classical clustering

Models	C1		C2		C3,		C4	
	Rules	Time	Rules	Time	Rules	Time	Rules	Time
Data using Fuzzy clustering	2/ 15	0.340	2/100	0.344	2/109	0.34	2/276	0.359
Classical clustering	2/15	0.344	2/99	0.344	2/108	0.344	2/271	0.360
W/O Clustering	74 Rules, 0.968 seconds							

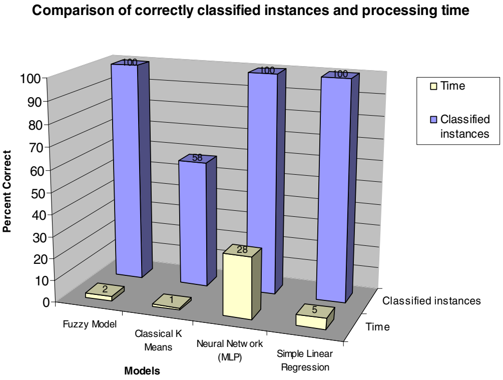


Fig. 4. Performance comparison among the models

Table 1 shows the comparison of the fuzzy, classical clustering and the non-clustered dataset. It is noted that the former performs faster on pattern discovery than the other methods as indicated by the processing time. The discovered best rules on each cluster indicate commonness of the rules obtained in the other clusters and the un-clustered dataset. Figure 4 shows that the accuracy of the fuzzy model which accounted to 100.0 % with classifying time of 2.0 seconds. It can be noted that for the same dataset, the classical k-means reveals faster processing time (1.0 sec.) but lower correctly classified instances (58%). For NN and SLR, it have values of (100%, 28 sec.) and (100%, 5 sec.), respectively. Then it follows that the membership accuracy is 1.0 for Fuzzy, NN and SLR while 0.58 for K-Means. This implies that the latter has a lower membership accuracy value than the other models. And this further implies that the fuzzy model shows higher MA and good classifying time.

5 Experimental Evaluations

The experiment was performed on the Internet usage dataset available at the UCI Knowledge Discovery on Databases (KDD) archive [6]. The dataset contains 70 attributes and 10,104 instances or observations. It comes from a survey conducted by the Graphics and Visualization Unit at Georgia Institute of Technology and it is about the "general demographics" of Internet users. The evaluation platforms used in the study were IBM compatible PC, Window OS, Java, Python and an open source Weka [14] machine learning algorithms for data mining.

5.1 Clustering and the Discovered Patterns

This procedure attempts to identify relatively homogeneous groups of cases based on selected characteristics. Table 2 shows that a total of 4 clusters had been identified and the group membership of each case is partially indicated in the same table. The table uses all the instances and classified using the fuzzy algorithm. A corresponding 4 clusters had been identified and the discovered rules based on 95 percent support threshold are also indicated in the same table. A 99% confidence had been set prior to the experiment. This implies that the same confidence for the best rules was obtained.

The result shows only the first five rules generated for each of the cluster. In cluster 1, the first rule means that V36 (computer platform) is associated to V62 (webpage creation) with support of 95% and confidence of 99%. The second rule means that V37 (primary language) is associated to V62 (webpage creation) with confidence of 99% while the third rule means that V36 (computer platform) and V37 (primary language) is associated to V62 (webpage creation) with confidence of 99%. The same fashion of explanation could be done to other rules.

It is essential to learn that there would be an improvement in processing time since the computation is based on chunks of data, i.e. processing of clustered instances. It is also interesting to note the difference of computing time as revealed by the graph below, showing the comparison of the original and clustered dataset.

Shorter processing time had been observed when computing for smaller clusters implying faster and ideal processing period than dealing with the entire dataset. The result of the processing time comparison is shown in Figure 7.

Table 2. Summary of cluster assignment and discovered rules

	Un-clustered Dataset	**Clustered Instances**			
Clusters→	All	1	2	3	4
Member Instances→	10,104	5,394	68	4,344	298
Best Rules Generated→	V60 ⇒V62 V36 V60 ⇒V62 V32 V60 ⇒V62 V32 V36 ⇒ V60 V62 V37 V60 ⇒V62	V36⇒ V62 V37 ⇒ V62 V36 V37 ⇒ V62 V32 ⇒ V62 V32 V36 ⇒ V62	V27⇒ V20 V20⇒ V27 V32⇒ V20 V20⇒ V32 V34⇒ V20	V36⇒ V62 V37⇒ V62 V36 V37⇒ V62 V32 ⇒V62 V32 V36⇒ V62	V62⇒ V36 V36⇒ V62 V32⇒ V36 V32⇒ V62 V37⇒ V36
Time to process rules (sec.)	10 .0	4.0	0.5	4.0	0.5

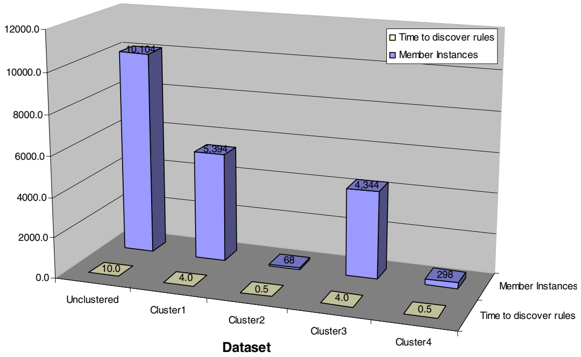


Fig. 5. Comparison of processing time

The result further implies that the blending of fuzzy clusters and association algorithm specifically isolate groups of correlated cases. This resulted to some partitions where we could conveniently analyze specific associations among clusters. In addition, the rules obtained per cluster indicated similarities of rules obtained for the entire dataset. It is imperative to generalize, although, not generally proven that since each cluster constitute higher correlations among the instances then the patterns obtained for such cluster will bring more meaning during analysis.

6 Conclusions

The preceding results were implemented using the approach shown in an example in section 4 and the model is tested on a dataset obtained from UCI database repository. The experiment reveals efficiency in relation to convenience and practicality of analyzing the results based on the discovered rules. It can be noted that the discovered best rules on each cluster indicated commonness of the rules obtained from the other clusters and the un-clustered dataset.

The results used the blending of the proposed fuzzy clustering and data mining algorithms. Higher clustering accuracy and better processing time have been observed using the proposed model. Shorter processing time had also been observed in computing for smaller clusters implying faster and ideal processing period than dealing with the entire dataset.

We have provided examples, performed experiments and generated results but more rigorous treatment maybe needed if dealing with more complex and other types of databases. Although, we were not able to rigorously prove the specific behaviors of rules obtained per cluster versus the entire dataset, it remains as the future task of this study. Other future investigations will include scalable fuzzy clustering and to carefully compare it with other existing intelligent system models.

Acknowledgement. Bobby D. Gerardo is a Faculty of the Institute of Information and Communications Technology, West Visayas State University, Iloilo City, Philippines. This paper was supported by Wonkwang University of Korea in 2005.

References

1. Lee, S.B., Lee T.S., Kim, H.J., and Le, Y.K.: Development of Traffic accident prediction Models with Intelligent System Theory, Lecture Notes in Computer Science, Vol. 3481, Singapore (2005) 880-888
2. Zadeh, L.A.: Fuzzy Sets, Information and Control, Vol. 8 (1965) 338-353
3. Kim, J.S., Kim, M.S., Noh, B.N.: A Fuzzy Expert System for Network Forensic, Lecture Notes in Computer Science, Vol. 3043, Assisi, Italy (2004) 175-182
4. Crestani, F. and Pasi, G.: Soft Information Retrieval: Applications of Fuzzy Set Theory and Neural Networks. In: N. Kasabov and R. Kozma, editors, Neuro-Fuzzy Techniques for Intelligent Information Systems, Physica Verlag (Springer Verlag), Heidelberg, Germany (1999) 287-315
5. Hollmen J.: Self Organizing Map, available at: http://www.cis.hut.fi/ ~jhollmen/ dippa/ node9.html (1996)
6. UCI Knowledge Discovery in databases, available at http://kdd.ics.uci.edu/
7. Chen, B., Haas, P., and Scheuermann, P.: A new two-phase sampling based algorithm for discovering association rules. Proceedings of ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (2002.)
8. Han, E. H., Karypis G., Kumar, V., and Mobasher, B.: Clustering in a high-dimensional space using hypergraph models. Available at: http://www.informatik uni-siegen.de/ ~galeas /papers/ general/ Clustering_in_a_High- Dimensional_ Space_ Using_ Hypergraphs _ Models_ 28 Han 1997b 29.pdf (1998)
9. Agglomerative Hierarchical Clustering. Available at http://www2.cs.uregina.ca/ ~hamilton/ courses/831/ notes/ clustering/ clustering.htm
10. Brule J. F.: Fuzzy Systems. Available at http://www.austinlinks.com/ Fuzzy/ tutorial.html
11. Saint-Paul, R., Raschia, G., and Mouaddib, N.: General Purpose Database Summarization, In proceedings of the 31st VLDB Conference, Trondleim, Norway (2005) 733-744
12. Dembele, D. and Kastner, P.: Fuzzy C-Means for Clustering Microarray Data, Journal of Bioinformatics, Volume 19, Number 8 (2003) 973-980
13. The Generic Fuzzy Clustering Algorithm, available at http:// polywww.in2p3.fr/ activities/ info/doc/ glast/ fc.htm#2 (2005)
14. Witten, I. and Frank, E.: Data Mining: Practical machine learning tools and techniques, 2nd Edition, Morgan Kaufmann, San Francisco (2005)

Metrics Design for Software Process Assessment Based on ISO/IEC 15504

Sun-Myung Hwang and Hee-Gyun Yeom

Department of Computer Engineering Daejeon University,
96-3 Yongun-dong, Dong-gu, Daejeon 300-716,
South Korea
{sunhwang, yeom}@dju.ac.kr

Abstract. In the current marketplace, there are maturity models, standards methodologies and guideline that can help an organization improve the way it does business. Software process assessment models, ISO/IEC 15504 and CMMI provide a tool to assess your organization's software development capability. Experienced assessors make these assessments. However these models don't supply systematic metrics for software process assessment. Therefore the assessors have used their subjective estimations for quantitative measurement in their software process assessment. This paper defines the basic metrics and presents the standard metrics in categories of process defined by ISO/IEC 15504 to assess software process quantitatively and objectively. In addition, presents an essential guideline to identify your organization's condition by suggesting a process maturity assessment metrics to apply the standard metrics to your organizations.

1 Introduction

The quality of a product depends on quality of a process is a known fact. Many industrial software organizations have put effort to improve their software process, which based on ISO/IEC 15504, CMMI. To improve the quality of software and their organization's software development capability and productivity, various approaches have been tried [3][11]. Process assessment enables to identify the process capability, and based on the resulted assessment you can expect an enhancement of the process by identifying your process strengths, weaknesses and risks and preventing them. This paper intends to present the methods of how to design standard metrics and apply them, on which you can assess the main process defined by an assessment model in order to measure an achievement of the process goal in their performing organization and an achievement of their activities quantitatively. In chapter 2, a representative software process capability assessment model will be introduced, in chapter 3, ISO/IEC 15504 process assessment metrics will be defined, process maturity assessment metrics will be presented in chapter 4, and the last chapter 5 will conclude this paper.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 909 – 916, 2006.

2 Software Process Capability Assessment Model

An organization having higher development maturity has better software process in the overall areas of their organization and they can implement software more consistently. Software process capability refers to the ability of the organization to produce these products predictably and consistently. A capability level is a set of process attributes that work together to provide a major enhancement in the capability to perform a process. Each level provides a major enhancement of capability in the performance of a process [1].

2.1 ISO/IEC 15504

Alike CMMI, ISO/IEC 15504 also assesses process maturity by identifying present process condition to support organization's process enhancement activities [10]. It defines a Process Assessment Model (PAM) that supports the performance of an assessment by providing indicators for guidance on the process purpose [2].

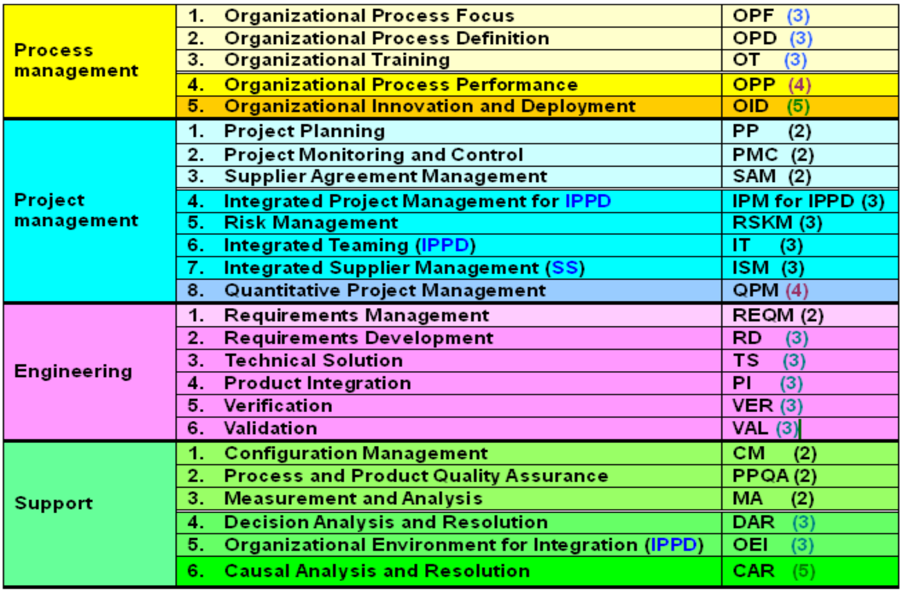
Table 1 shows the process defined by ISO/IEC 15504.

Table 1. ISO/IEC 15504 Process

PRIMARY Life Cycle Processes		ORGANIZATIONAL Life Cycle Processes
1. Acquisition Group ACQ.1 Acquisition Preparation ACQ.2 Supplier selection ACQ.3Supplier monitoring ACQ.4Customer acceptance	2. Supply Group SPL.1 Supplier tendering SPL.2 Contract agreement SPL.3 Software release SPL.4Software acceptance	1. Management Group MAN.1 Organizational alignment MAN.2 Organization management MAN.3 Project management MAN.4 Quality Management MAN.5 Risk Management MAN.6 Measurement
3. Engineering Group ENG.1 Requirement elicitation ENG.2 System requirement analysis ENG.3 System architectural design ENG.4 Software requirement analysis ENG.5 Software design ENG.6 Software construction ENG.7 Software integration ENG.8 Software testing ENG.9 Software installation ENG.10 System integration ENG.11 System testing ENG.12 System & software maintenance		2. Process Improvement Group PIM.1 Process establishment PIM.2 Process assessment PIM.3 Process improvement 3. Resource & Infrastructure Group RIN.1 Human resource management RIN.2 Training RIN.3 Knowledge management RIN.4 Infrastructure 4. Reuse Group REU.1 Asset management REU.2 Reuse program management REU.3 Domain engineering
4. Operation group OPE.1 Operational use OPE.2 Customer support		
SUPPORTING Life Cycle Processes		
1. Configuration control Group CFG.1 Documentation Management CFG.2 Configuration Management CFG.3 Problem Management CFG.4 Change Request Management	2. Quality Assurance Group QUA.1 Quality assurance QUA.2 Verification QUA.3 Validation QUA.4 Joint review QUA.5 Audit QUA.6 Product Evaluation	

2.2 CMMI

SW-CMM was sunset because SEI decided not to supply SW-CMM used for last 10 years till end of 2003 any more and to distribute only CMMI after 2005[4]. CMMI provides a framework for introducing new disciplines about systems engineering and software engineering as needs arise [12]. CMMI process can be simplified as a Figure 1.



Process management	1.	Organizational Process Focus	OPF (3)
	2.	Organizational Process Definition	OPD (3)
	3.	Organizational Training	OT (3)
	4.	Organizational Process Performance	OPP (4)
	5.	Organizational Innovation and Deployment	OID (5)
Project management	1.	Project Planning	PP (2)
	2.	Project Monitoring and Control	PMC (2)
	3.	Supplier Agreement Management	SAM (2)
	4.	Integrated Project Management for IPPD	IPM for IPPD (3)
	5.	Risk Management	RSKM (3)
	6.	Integrated Teaming (IPPD)	IT (3)
	7.	Integrated Supplier Management (SS)	ISM (3)
	8.	Quantitative Project Management	QPM (4)
Engineering	1.	Requirements Management	REQM (2)
	2.	Requirements Development	RD (3)
	3.	Technical Solution	TS (3)
	4.	Product Integration	PI (3)
	5.	Verification	VER (3)
	6.	Validation	VAL (3)
Support	1.	Configuration Management	CM (2)
	2.	Process and Product Quality Assurance	PPQA (2)
	3.	Measurement and Analysis	MA (2)
	4.	Decision Analysis and Resolution	DAR (3)
	5.	Organizational Environment for Integration (IPPD)	OEI (3)
	6.	Causal Analysis and Resolution	CAR (5)

Fig. 1. CMMI Process

3 ISO/IEC 15504 Process Assessment Metrics

3.1 Basic Metrics Definition

To maximize the effect of organization management a metrics is required which measuring the process performance of the software project quantitatively. For an organization, to be qualified by ISO/IEC 15504 or assessed Level 4 or above by CMMI quantitative process management must be followed and all process performance (basic performance, management performance) has to be measurable [1].

The measures in quality process management consist of the size of product, deadline, man month, cost, resource, modification, risk and defect, which are basic elements for the metrics [7].

3.2 ISO/IEC 15504 Standard Metrics Design

Based on the basic performance of 48 number sub-processes categorized in ISO/IEC 15504 a standard metric definition can be defined on the basis of basic metrics[8][9].

Table 2. Basic Metrics

Category	Metrics name	formulas
Process	Compliance of planed process	(number of executed process /number of planned process)*100
	Progress ratio compliance of planned	(actual ratio of progress/planned ratio of progress)*100
MM	MM	(actual mm/planned mm)*100
Cost	Cost ratio	(actual cost/planned cost)*100
Productivity	Productivity in each process	Analysis: number of requirement /mm Design: number of design item/ mm Implementation: (fp or loc)*mm
Size	Compliance of product size	(actual size/planed size)*100
Resource	Compliance of Computer resource	(actual computer resource/estimated computer resource)*100
Requirement	Ratio of CR	(number of changed requirement/number of initial requirements)*100
Quality	Ratio of risk occurrence	(number of realized risk/number of fined risk)*100
	Ratio of fault remove	(number of removed fault/number of discovered fault)*100

Table 3. Standard Metrics Definition-Example

Metric	Actual Size Ratio vs Expectation				
Metric ID	M001	Category	Size	ISO/IEC 15504	CUS.2 Supply process
Reporting Time	plan-close	Lower bound	50	Upper bound	150
Formulas	A/B*100			Unit	%
Information of measured value					
A	Actual Size				
B	Expectation Size				
Guidance for application					
Outline	Measure the accuracy of the project size assessment. Size unit is measured by KSLOC(1,000 Source Line of Code), include an annotation and except the blank line. Whenever the expectation size can be estimated or re-estimated, it must be measure. The actual size must be measured after the test level.				
Analysis Method	Whenever it is re-estimated, analyze the addition and reduction situation of the size, compare and analyze the actual size with the estimated value after coding.				
Indicator	A graph of broken line				
Explaining as the Result	85 =< Ratio =< 115 : The appropriate estimating for the size Ratio>115: The underestimating for the size. This ratio may made the cost and schedule increase, which measure to keep the schedule through the changing cost and additional MM as reschedule. Ratio<85: The overestimating for the size. Arrange the unnecessary cost and manpower as reschedule.				

The components of metric definition are metrics name, metrics ID, category belonging to basic metrics, related ISO/IEC 15504 processes, reporting time of metrics measured, allowable maximum and minimum scope of relevant measured values, calculation for measured value, value unit and related measures, method to analyze related metrics semantically, indicators to express those meaning effectively, result analysis showing the meaning of measured results and according to those results indications and procedures need to be taken[8].

3.3 ISO/IEC 15504 Group Process Measurement Metrics

3.3.1 Engineering Group Process Measurement Metrics

To measure Engineering Group process 23 numbers of metrics are defined [6]. By using one of those metrics, ENG.1 Requirement Elicitation we define a metrics shows an equivalent rate of requirements allocated to software and system requirement list checking if there is enough system resource for balance. Table 4 below show the metrics definition.

Table 4. Requirement Elicitation Process Measurement Metrics Definition

Metric	Compliance of System Requirement Item				
Metric ID	ENG01	Category	Requirement	ISO/IEC 15504	ENG.1 Requirement Elicitation
Reporting Time	Before ENG.2	Lower bound	50	Upper bound	100
Formulas	A/B*100			Unit	%
Information of measured value					
A	a number of software requirement analysis				
B	a number of software requirement in the system requirement analysis list				
Guidance for application					
Outline	Indicate system requirement for balancing with assigned requirement in software.				
Analysis Method	(a number of software requirement analysis / a number of software requirement in the system requirement analysis)*100				
Indicator	A graph of broken line				
Explaining as the Result	The more ratio approach to 100, the more system requirement is balanced.				

3.3.2 Configuration Control Group Process Measurement Metrics

To measure Configuration Control Group process 4 numbers of metrics are defined.

The metrics of verified configuration item rate in comparison of all is shown in Table 5.

Table 5. Configuration Management Process Measurement Metric Definition

Metric	Verified Configuration Item Rate				
Metric ID	CFG02	Category	Process	ISO/IEC 15504	CFG.2 Configuration Management
Reporting Time	Before CFG.3	Lower bound	50	Upper bound	100
Formulas	A/B*100			Unit	%
Information of measured value					
A	verified configuration items				
B	total configuration items				
Guidance for application					
Outline	It shows the observance degree of the configuration management work which is completed				
Analysis Method	(a number of verified configuration items/ a number of total configuration items)*100				
Indicator	A graph of broken line				
Explaining as the Result	it confronts to the configuration item possibility of being defined and as ratio of the item which is verified in 100 near recording process accomplishment it is high the meaning				

4 Process Capability Assessment Metrics

To measure a process capability resulted in SPA (Software Process Assessment) quantitatively is very difficult. The properties of process which determining the process capabilities are formed with qualitative actions. For the better quantitative assessment to a given process capability we suggest PCM (Process Capability Metrics).

Through the indicators measuring achievement of PA in each capability levels we set questions, which can check those indicators to identify each process capabilities.

PCMT (process capability metric table) is defined for assessors can input assessment point into each question. It is shown in Figure 2.

For capability assessment, PA, GPI, GRI, GWPI and so forth are defined in its level and assessors fill each items with assessment point by gathering each process performance achievement. GPI item is a mandatory item to be filled. Others can be optional by assessor consultation.

Making a PCMT table, you can gather relevant PCM result when you consider the following conditions:

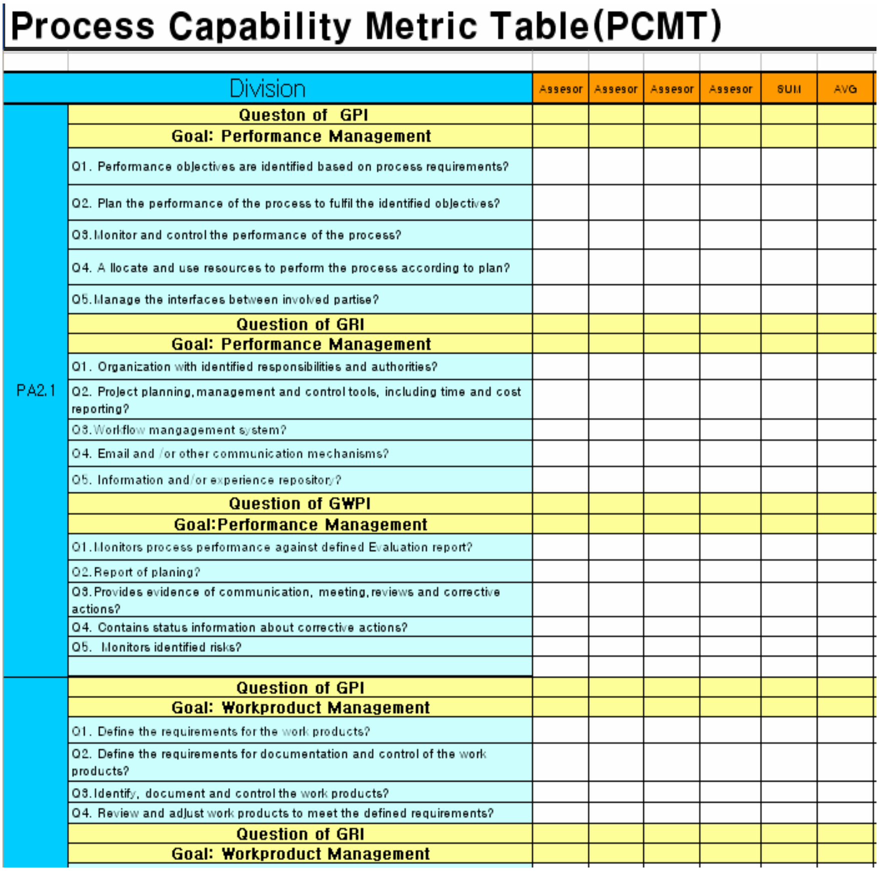
Assessment has to cover one step higher target then your objective target

GPI item is a mandatory item to be filled. Others can be optional by assessor consultation

An average assessment point by many assessors can make assessment for same item.

The defined PCM for process capability is

$$PCM = 5 \times \left(\sum_{i=1}^{n} \frac{Q_i}{100}\right) \times \frac{1}{n}$$



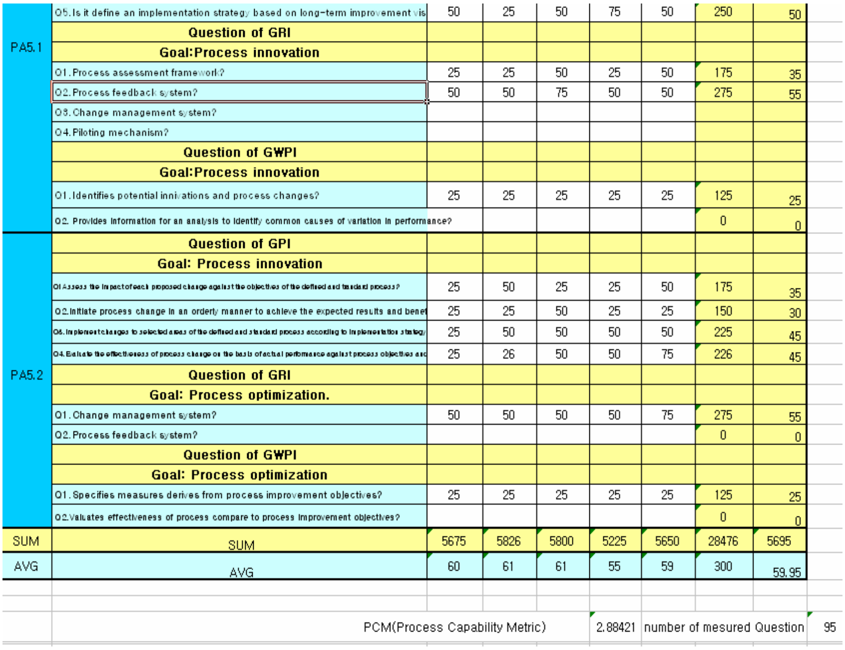
Process Capability Metric Table(PCMT)

	Division	Assesor	Assesor	Assesor	Assesor	SUM	AVG
PA2.1	**Queston of GPI**						
	Goal: Performance Management						
	Q1. Performance objectives are identified based on process requirements?						
	Q2. Plan the performance of the process to fulfil the identified objectives?						
	Q3.Monitor and control the performance of the process?						
	Q4. A llocate and use resources to perform the process according to plan?						
	Q5.Manage the interfaces between involved partise?						
	Question of GRI						
	Goal: Performance Management						
	Q1. Organization with identified responsibilities and authorities?						
	Q2. Project planning, management and control tools, including time and cost reporting?						
	Q3.Workflow mangagement system?						
	Q4. Email and /or other communication mechanisms?						
	Q5. Information and/or experience repository?						
	Question of GWPI						
	Goal:Performance Management						
	Q1.Monitors process performance against defined Evaluation report?						
	Q2.Report of planing?						
	Q3.Provides evidence of communication, meeting, reviews and corrective actions?						
	Q4. Contains status information about corrective actions?						
	Q5. Monitors identified risks?						
	Question of GPI						
	Goal: Workproduct Management						
	Q1. Define the requirements for the work products?						
	Q2. Define the requirements for documentation and control of the work products?						
	Q3.Identify, document and control the work products?						
	Q4. Review and adjust work products to meet the defined requirements?						
	Question of GRI						
	Goal: Workproduct Management						

Fig. 2. Process Capability Metric Table (PCMT)

5 Conclusion

In this paper we suggest the metrics in each processes enables organizations to predict a direction for active process enhancement and to quantize present process condition specifically and to identify if the goal of process can achieve. This objective process metrics based on ISO/IEC 15504, which has not been introduced in previous process assessment models, can be expected to measure process capability and to identify the risk, problems, and condition of process performance by using these metrics.



PA5.1	O5. Is it define an implementation strategy based on long-term improvement vis	50	25	50	75	50	250	50	
	Question of GRI								
	Goal:Process innovation								
	O1. Process assessment framework?	25	25	50	25	50	175	35	
	O2. Process feedback system?	50	50	75	50	50	275	55	
	O3. Change management system?								
	O4. Piloting mechanism?								
	Question of GWPI								
	Goal:Process innovation								
	O1. Identifies potential innivations and process changes?	25	25	25	25	25	125	25	
	O2. Provides information for an analysis to identify common causes of variation in performance?						0	0	
PA5.2	Question of GPI								
	Goal: Process innovation								
	OI Assess the impact of each proposed change against the objectives of the defined and tandard process?	25	50	25	25	50	175	35	
	O2. Initiate process change in an orderly manner to achieve the expected results and benet	25	25	50	25	25	150	30	
	O3. Implement changes to selected areas of the defined and standard process according to implementation strategy	25	50	50	50	50	225	45	
	O4. Evaluate the effectiveness of process change on the basis of actual performance against process objectives and	25	26	50	50	75	226	45	
	Question of GRI								
	Goal: Process optimization.								
	O1. Change management system?	50	50	50	50	75	275	55	
	O2. Process feedback system?						0	0	
	Question of GWPI								
	Goal: Process optimization								
	O1. Specifies measures derives from process improvement objectives?	25	25	25	25	25	125	25	
	O2. Valuates effectiveness of process compare to process improvement objectives?						0	0	
SUM	SUM	5675	5826	5800	5225	5650	28476	5695	
AVG	AVG	60	61	61	55	59	300	59.95	
	PCM(Process Capability Metric)				2.88421	number of mesured Question			95

Fig. 3. PCMT Application Method

References

[1] H.M.Kim, S.M.Hwang, "A Study on Metrics for supporting the Software Process Improvement based on SPICE", SERA04, Los Angeles, 2004

[2] G.J.Kim,"International standard for SPICE S/W process assessment", Software Engineering Review, Vol.10, No.4, pp.58-71, 1997

[3] Sun-Myung Hwang, "Analysis of Relationship among ISO/IEC 15504,CMM, and CMMI",SERA03, SanFrancisco, 2003

[4] Pankaj Jalote, CMM in Practice, SEI Series in Software Engineering, 2000

[5] Dennis M.Ahern, Aaron Clouse, and Richard Turner, CMMI distilled, SEI Series in Software Engineering, 2001

[6] Azuma, "Software Quality Evaluation System: Quality Models Metrics and Processes - International Standards and Japanese Practice", Information and Software Technology, 1996

[7] Kii-Won Song, "Research about confidence verification of KPA question item through SEI Maturity Questionnaire's calibration and SPICE Level metathesis modeling", SERA03, San Francisco, 2003

[8] "Software Design Method enhanced by Appended Security Requirements", LNCS, 3331, pp.578-585, 2004

[9] "A Design of Configuration Management Practice and CMPET in CC Based on SW Process Improvement Activity", LNCS, 3043, pp.481-490, 2004

[10] "A Study on Metrics for Supporting the Software Process Improvement based on SPICE

[11] ISO/IEC 14598-1,2,3,4 Information Technology Software Product Evaluation, 1999

[12] ARC. 2000. Assessment Requirements for CMMI, Version 1.0 CMU/SEI-2000-TR-011. Software Engineering Institute, Carnegie Mellon University, Pittsburgh: PA.

A Quantitative Evaluation Model Using the ISO/IEC 9126 Quality Model in the Component Based Development Process

Kilsup Lee and Sung Jong Lee

Dept. of Computer & Information, Korea National Defense University,
205, Soosaek, Eunpyung, Seoul, 122-875, Republic of Korea
{gislee, ljc}@kndu.ac.kr
http://www.kndu.ac.kr

Abstract. Recently, software quality evaluation based on ISO/IEC 9126 and ISO/IEC 14598 has been widely accepted in various areas. However, these standards for software quality do not provide practical guidelines to apply the quality model and the evaluation process of software products. Thus, we present a quantitative evaluation model using the ISO/IEC 9126 quality model in the Component Based Development (CBD) process. Particularly, our evaluation model adopts a quantitative quality model which uses the weights of quality characteristics obtained through carefully selected questionnaires for stakeholder and Analytic Hierarchical Process (AHP). Moreover, we have also examined the proposed evaluation model with applying the checklists for the artifacts of the CBD to a small-scale software project. As a result, we believe that the proposed model will be helpful for acquiring the high quality software.

1 Introduction

Recently, software quality evaluation based on the ISO/IEC 9126 [1] software product quality model and the ISO/IEC 14598 [2] software product evaluation process have been widely accepted in various areas. However, these standards do not provide practical guidelines to apply the quality model and the evaluation process of software products. This fact makes it difficult to use these standards in real software projects.

In the acquisition process, requirements of quality evaluation are sometimes missing in some Request for Proposals (RFPs) or Proposals. Most activities of quality management have been focused on detection and correction of defects through a field overseeing, review meetings, tests, and audits. The standards for software quality are not considered in most quality assurance activities in real projects. Thus, the inadequate management of software quality causes un- expected defects in test and operation phases. In a worst case, it will result in the failure of delivery or the increase of maintenance cost.

In software quality management, there are extensive research works on software quality model and evaluation process. The issues focused are metrics on COTS (Commercial Off The Shelf) based systems [3], metrics and models for cost and

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 917 – 926, 2006.

quality of component-based software [4], quality models in software package selection [5], web application quality with WebQEM [6], and ontology for software metrics and indicators [7]. However, the research works to evaluate artifacts from software development process using the standard quality model and evaluation process are not well known.

The Ministry of National Defense of Republic of Korea (ROK- MND) announced the Component Based Development (CBD) process for defense software in 2005 [8]. The ROK-MND has a goal to adopt the CBD in research and development projects. But the quality management was not the scope of the study on the CBD, though it is as important as development process. Thus, we present a software quality evaluation model to support the Component Based Development (CBD) of the ROK-MND. Particularly, the quantitative evaluation model is refined from the ISO/IEC 9126 quality model.

The rest of this paper is organized as follows. In section 2, we survey ISO/IEC 9126, ISO/IEC 14598, and the CBD of ROK-MND as the related works. In section 3, we present a quantitative evaluation model with weights of quality characteristics which are obtained by carefully selecting questionnaires for the stakeholder and Analytic Hierarchical Process (AHP) [9]. In section 4, we describe a case study for our proposed approach through a trial evaluation of the CBD artifacts from a small-scale project. In section 5, we conclude the results of our research and discuss the further works.

2 Related Works

In this section, we describe the ISO/IEC 9126 software product quality model, the ISO/IEC 14598 software product evaluation process and the CBD of the ROK-MND. First of all, the quality model in the ISO/IEC 9126 has been developed through the models from Boehm [10], McCall [11], Evans [12], etc.. A quality model gives a general purpose framework for approaches to development process quality, product quality, the life cycle and items to be evaluated. A software evaluation process uses a quality goal, a quality model, quality characteristics and their metrics in order to evaluate software products. Now, we focus on the standard quality model of ISO/IEC 9126 for our works.

The ISO/IEC 9126 describes software product quality and consists of four parts such as quality model and metrics with respect to external attributes, internal attributes, and quality in use. The quality characteristics are classified into subcharacteristics. This standard also describes six quality characteristics and guidelines for its use, which might be useful not only for evaluating a software product but also for defining quality requirements and other usage. The 6 characteristics are as follows: functionality, reliability, usability, efficiency, maintainability, and portability. Moreover, they are refined into 27 subcharacteristics such as suitability, accuracy, and so on.

The software quality evaluation process in ISO/IEC 14598 is composed of several phases such as establishing evaluation requirements, specifying evaluation, designing an evaluation plan, and executing evaluation. In the establishment phase of quality requirements, the degree of quality requirement is defined using quality characteristics and their available subcharacteristics, which are necessarily defined before development. And the quality requirements of a product can be differently applied to its components according to the properties of the components.

Table 1. The Component Based Development (CBD) process

Phases	Activities	Artifacts
Analysis (1R)	Requirement definition (1R1)	Glossary
		Requirement specification
	Architecture definition (1R2)	System architecture definition
	Requirement analysis (1R3)	Usecase specification
		Class specification
Design (2D)	Preliminary design (2D1)	Component catalog
		Component architecture definition
		Interface interaction specification
		Interface specification
		Component specification
		Data design
	Detailed design (2D2)	Component design
		Class specification
Implementation and Test (3T)	Test preparation (3T1)	Test plan
		Component test design
	Implementation (3T2)	Physical database
		Component code
		Component test design
		User interface code
	Integration test (3T3)	Integration test design
		Integration test result
Delivery (4S)	System analysis (4S1)	System installation plan
		System installation report
	Acceptance report (4S2)	Training report

In the specification phase of an evaluation, appropriate metrics should be provided not only for product characteristics but also for interactions between a product and its environment. Moreover, a measurement scale, ranks and final criteria for an evaluation should be provided. In the design phase of evaluation, we need to produce an evaluation plan which describes the evaluation methods and the schedule of the evaluation actions. Also it should be consistent with the measurement plan. Finally, in the execution phase of an evaluation, actual measurement is performed. The results of the measurement are compared with criteria and assessed in an overall point of view to make a final decision.

The CBD of ROK-MND is composed of 4 phases, such as analysis, design, implementation and test, and delivery. Each phase is decomposed into activities and artifacts. Table 1 shows the CBD process for a small-scale development project. Here, we use some identifiers to represent phase order, phase acronym, and activity number (e.g., 1R, 2D, 3T, 4S, etc.). An identifier of the full CBD process consists of phase order, phase acronym, activity number, task number, and artifact order. For example, the identifier, *2D11a*, represents an artifact, *component catalog,* from a task, *component identification,* in an activity, *preliminary design,* in the *design* phase.

3 A Qualitative Quality Evaluation Model

In this section, we present a quantitative quality evaluation model in order to evaluate a software product based on ISO/IEC 9126 and to compare the user's quality goal with the measured value. The overall model explains the relationship among user's quality goal, measured quality values, quality characteristics, subcharacteristics, and their external and internal metrics. The internal metrics denote the software attributes, while the external metrics denotes software behaviors.

3.1 An Overall Model for Quality Evaluation

In the requirement phase, users establish their quality goal from needs with respect to quality in use. The needs of a system determine the external quality goal in the specification phase and the internal quality goal in the design and development phases. The external quality goal determines the internal quality goal. As the quality in use of a system is measured through external metrics in the operation phase, the external quality is measured through external metrics in the system integration and testing phases. But the internal quality is measured at the design and development phases.

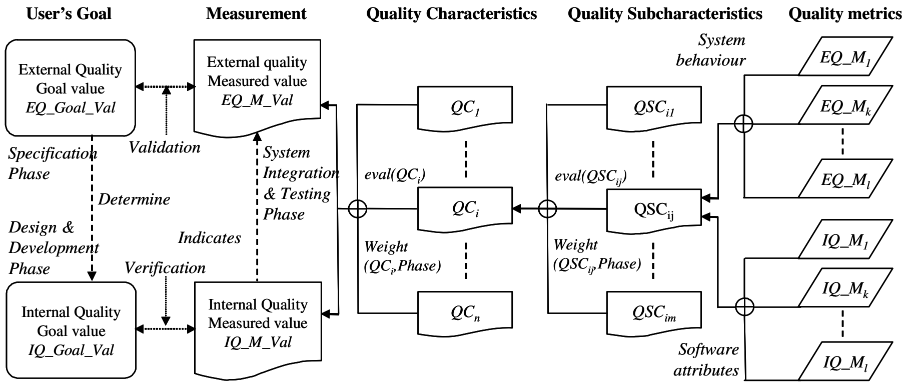


Fig. 1. The quantitative quality evaluation model

On the contrary, the internal quality indicates the external quality and the external quality indicates the quality in use. The quality goal values can be the criteria for the artifacts under evaluation. Fig. 1 shows the quantitative quality evaluation model.

In an artifact evaluation, we select quality characteristics, subcharacteristics, metrics and their weights which reflect the output from the establishment phase of evaluation requirements. First of all, we describe our evaluation approach using internal metrics. The evaluated value *eval(P,* QSC_{ij}*)* is as follows, where QSC_{ij} is the

j^{th} subcharacteristic of the i^{th} characteristic of a product *P*, IQ_M_k is the k^{th} internal quality metric, $1 \leq k \leq l$, and *l* is the number of internal metrics for a product *P*.

$$\sum_{k=1}^{l} \frac{eval(P, QSC_{ij}, IQ\_M_k)}{l} \qquad (1)$$

And the evaluated value *eval(P,* QC_i*)* is as follows, where QC_i is the i^{th} characteristic of a product *P*, *weight(*QSC_{ij}*, Phase)* is the weight factor of the subcharacteristic of QSC_{ij} in the development phase *Phase*, $1\leq j \leq m$, $\Sigma_{j=1,m}$ *weight(*QSC_{ij}*, Phase) = 1,* and *m* is the number of subcharacteristics of a characteristic QC_i.

$$\sum_{j=1}^{m} (eval(P, QSC_{ij}) \times weight(QSC_{ij}, Phase)) \qquad (2)$$

As a result, the internal quality measurement value *IQ_M_Val* through internal metrics is calculated as follows, where *weight(*QC_i*, Phase)* is the weight factor of the characteristic of QC_i in the development phase *Phase*, $1\leq i\leq n$, $\Sigma_{i=1,n}$ *weight(*QC_i*, Phase) = 1*, and *n* is the number of characteristics of a product *P*.

$$\sum_{i=1}^{n} (eval(P, QC_i) \times weight(QC_i, Phase)) \qquad (3)$$

On the other hand, the external quality measurement value *EQ_M_Val* is measured through external metrics as the internal quality evaluation value. Moreover, we can estimate *EQ_M_Val* from *IQ_M_Val.*

3.2 Weights of Quality Characteristics

In this clause we present the result of our questionnaire to determine weights of quality characteristics. We have made up questions for 50 people. They are carefully selected to include personnel concerned with entire life cycle, stakeholder, careers of people and the reliability of answers. The procedure to obtain the weights of quality characteristics is as follows. First, we analyze the respondents. Next, we analyze the weights according to the respondents. And then, we extract the weights of quality characteristics according to the phases of software development life cycle. Finally, we perform similar work to obtain weights of quality subcharacteristics.

The respondents are composed of four groups such as 17 acquisition managers (34%), 12 developers (24%), 11 users (22%) and 10 maintenance engineer (20%), and work for governmental organization, research institute for information technology, in-house development division of a government organization, and a corporation. They also have careers related with software such as 13 people (26%) with more than 10 years, 11 people (22%) between 7 and 9 years, 14 people (28%) between 3 and 6 years, and 12 people (24%) with less than 3 years of experience.

We have also inquired the recognition degree of respondents on importance of quality evaluation and suitability of quality evaluation using quality characteristics. As a result, 44 respondents (88%) have answered that software quality evaluation is important. And 37 people (74%) have said that quality evaluation using quality characteristics is suitable. Moreover, we also found that 43 people (86%) understand the contents of ISO/IEC 9126. Hence, we believe that the data collected is reliable considering the structure and capability of the respondents.

We have analyzed the collected data using AHP[9] to analyze weights of quality characteristics in software development life cycle. The Expert Choice is used as a tool to support AHP and the MS-Excel is used to process general data such as formation and careers of respondents, and the degree of recognition of quality evaluation. Particularly, AHP takes advantage of the reliability and consistency of analysis regardless of the size of samples.

Through AHP analysis, we have extracted the weights of quality characteristics in accordance with software development life cycle. The weight of a quality characteristic is based on phases such as planning, design, coding, testing, and delivery. Table 2 shows a weight profile of quality characteristics. Similarly, weights of subcharacteristics are also provided through AHP analysis. These weight values are just reference data to set up weights of quality characteristics. Evaluators can adjust these weights for their applications. Moreover, these weights can be refined according to applicable domain. Thus, it will produce various profiles of weights.

Table 2. A weight profile of quality characteristics

Characteristics	Planning	Design	Coding	Testing	Delivery
Functionality	0.169	0.164	0.211	0.165	0.126
Reliability	0.200	0.193	0.168	0.197	0.159
Usability	0.188	0.174	0.151	0.182	0.140
Efficiency	0.167	0.168	0.165	0.151	0.160
Maintainability	0.136	0.158	0.156	0.158	0.238
Portability	0.140	0.143	0.149	0.146	0.177

4 A Case Study

The proposed approach is applied to a small-scale project which has the development period of 100 days after the contract and which develops a web-based system to assure interoperations between a legacy system and a new system. The small project adopts the CBD. In the evaluation requirement specification phase, the artifacts under evaluation are identified. Then, quality characteristics and their subcharacteristics are selected to meet the evaluation goal of the subjective artifact. And then in the evaluation specification phase, proper quality metrics for subcharacteristics are selected carefully.

In the trial evaluation, we have developed 15 checklists for artifacts of the CBD such as requirement specification, usecase specification, and so on. A checklist can be created by the following steps during the software quality evaluation process, and also incorporates the mechanism of the quantitative quality evaluation model described in the clause 3.2. A checklist example for a requirement specification of the CBD is depicted in Table 3.

This checklist has the fields, quality characteristics, subcharacteristics, inputs A and B, measurement formula, the measured value of a metric, the weight value, the measured value of a characteristic and the total measured value. Here, if we assign input values, we can obtain the result instantly through the measurement formula. The measured value is also obtained through multiplying the weight value by the measured value.

Table 3. A checklist example for a requirement specification of the CBD. Ch(Characteristic), Subch(Subcharacteristic), m(metric), w(weight), M(Measured value).

Ch, Subch, m			Check item	w, M	Score		
Total quality							
Functionality				0.169		0.853	
	Suitability			0.050			0.871
		Functional adequacy	X = 1 - A1 / B1	0.825			
			A1 = Number of functions in which problems are detected in evaluation	14			
			B1 = Number of functions checked	80			
		Functional implementation completeness	X = 1 - A1 / B2	0.844			
			A1 = Number of functions in which problems are detected in evaluation	14			
			B2 = Number of functions described in requirement specifications	90			
		Functional implementation coverage	X = 1 - A2 / B2	0.944			
			A2 = Number of functions which is not implemented	5			
			B2 = Number of functions described in requirement specifications	90			
		......	......	......			

The rating level established is classified into five levels such as excellent, good, marginally acceptable, marginally unacceptable, and poor. And the rating level is related with score range from 0 to 1. These rating levels are classified into two rating categories. Table 4 shows the established rating levels.

In the evaluation plan design (E3D) phase, we have established a brief evaluation plan with the checklists for the selected artifacts of the CBD. This plan includes the information on schedule, place, and method to achieve the objectives of the plan. Next, in the evaluation execution (E4E) phase, we have measured the value for each metric in a checklist. And then, the measured value is evaluated using the rating levels shown in Table 4.

We have conducted the trial evaluation using the checklists for four artifacts of the CBD with respect to 6 characteristics. Fig. 2 depicts the result of the trial evaluation.

The measured value for the functionality of the class specification shows the highest value of 45%. The other values are distributed below 42%. Thus, the overall quality value marks low quality in the initial phase of the CBD. Moreover, we can find that the measured values for reliability, maintainability, and portability characteristics are 0%. This evaluation result is unsatisfactory to the assessment criterion 70% and even to the required quality goal at the requirement phase 80%. Thus, the overall artifacts have been rejected in the trial evaluation.

Table 4. The established rating levels

Rating levels	Score range	Categories
Excellent	Above 0.800	Satisfactory
Good	0.700 ~ 0.799	
Marginally acceptable	0.600 ~ 0.699	
Marginally unacceptable	0.500 ~ 0.599	Unsatisfactory
Poor	Below 0.500	

Through the result in Fig. 2, we have some findings as follows. First of all, user does not specify requirements properly in the initial phase. The quality score of requirements specification is 2.9% ~ 28.2% and that of Usecase specification is 13.2% ~ 43.2%. And most non-functional requirements show 0%. And this project is focused on implementation due to the very short duration, 100 days. Thus, the artifacts are managed loosely.

Here, our model shows the possibility of practical use in real projects. The checklist has been used effectively to find defects, shortages, and missing requirements in the early phase of the development process. And we can obtain a positive effect such that the defects can be corrected promptly using the result of the trial evaluation.

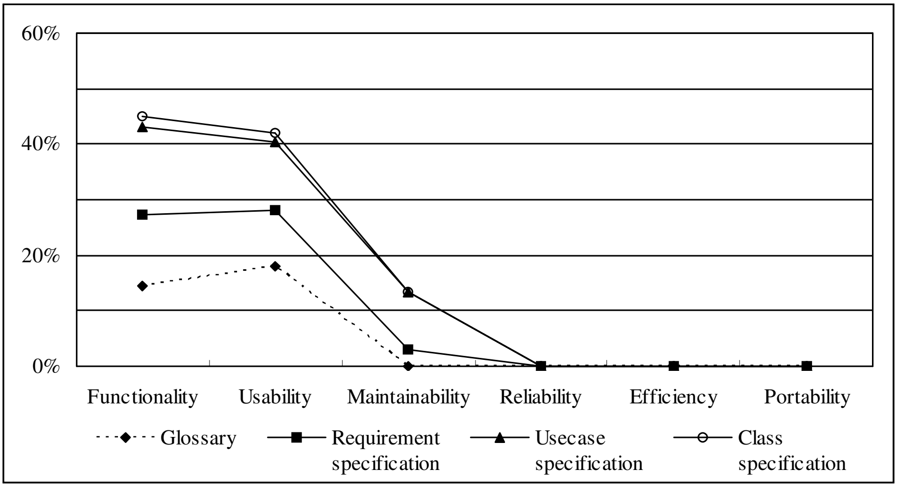


Fig. 2. The result of a trial evaluation

Furthermore, we can find that the requirements are not specified properly. Only functional requirements and the usability as a non-functional requirement are specified. Moreover, important non-functional requirements such as reliability, efficiency, and portability are missed. We can estimate that the requirements under evaluation are specified by novices without involving requirement experts. Thus, project manager can consider additional training or involving requirement experts.

Meanwhile, we found some complementary points as follows. First, we need to assure the relationship between the metrics and subjective artifacts. Particularly, it is not easy to select proper metrics among 68 internal metrics and 178 external metrics. Second, we need to develop various profiles for checklists according the applicable areas such as web-based, embedded, distributed software.

5 Conclusions

Recently, the importance of high quality software development has become more important than ever. We need to manage quality of software continuously throughout software development life cycle. Most previous research works [3]-[7] on software quality evaluation are focused on COTS-based software or deliverable software products with quality model and metrics. However, this paper has presented a quantitative quality evaluation approach with respect to the Component Based Development (CBD) methodology of ROK-MND. We have incorporated the ISO/IEC 9126 quality model and the ISO/IEC 14598 software product evaluation process into our proposed approach. The standard quality model includes quality characteristics, subcharacteristics, and quality metrics.

Particularly, the proposed quality evaluation approach uses the quantitative quality model with weights of quality characteristics and subcharacteristics based on questionnaires given to the expert groups which include all personnel in software development life cycle. The data from questionnaire have been processed through Analytic Hierarchical Process (AHP) technique. The weights of quality characteristics are tailored according to applicable domain by expert group and then they are merged into the quantitative quality evaluation model. The quantitative quality model is incorporated into the software quality evaluation process for the CBD.

We have also shown the viability of the proposed quality evaluation approach through a case study. The result shows the possibility of practical use in real projects. The evaluation model provides checklists supported by standard metrics. Hence, we can find missing requirements in the early phase of a development process. And we can differentiate software products various classes using the proposed evaluation process.

However, we need further works for general application to real projects. We require more elaboration on finding the appropriate size of software with respect to the overhead in preparing for an evaluation. And we need to assess the effectiveness by applying several types of software products (e.g., embedded, commercial, and developed software) and to elaborate on the metrics and their weights continuously.

References

1. ISO/IEC 9126-1: Information Technology - Software Product Quality - Part 1: Quality Model. ISO/IEC JTC1/SC7/WG6 (1999)
2. ISO/IEC 14598-1: Information Technology - Software Product Evaluation - Part 1: General Overview. ISO/IEC JTC1/SC7 (1998)
3. Sedigh-Ali, S., Ghafoor, A., Paul, R.A.: Software Engineering Metrics for COTS-Based Systems. IEEE Computer Magazine, May, (2001) 44-50
4. Sedigh-Ali, S., Ghafoor, A., Paul R.A.: Metrics and Models for Cost and Quality of Component-based Software. In: Proc. of 6th ISORC'03, IEEE Computer Society, May, (2003) 149-155
5. Franch, X., Carvallo, J.P.: Using Quality Models in Software Package Selection. IEEE Software, 20(1), (2003) 34-41
6. Olsina, L., Rossi, G.: Measuring Web Application Quality with WebQEM. IEEE MultiMedia, (2002) 20-29
7. Martin, M.A., Olsina, L.: Towards an Ontology for Software Metrics and Indicators as the Foundation for a Cataloging Web System. In: Proc. of the First Latin American Web Congress (LA-WEB 2003), IEEE Computer Society, (2003) 103-114
8. Ministry of National Defense: A Defense Methodology of Component Based Development. Republic of Korea (2005)
9. Saaty, T.L.: The Analytic Hierarchy Process. McGraw-Hill New York (1980)
10. Boehm, B.W., Brown, J. R., Lipow, M.: Quantitative evaluation of software quality. In: Proc. of the 2nd International Conference on Software engineering (1976) 592-605
11. McCall, J.A., Markham, D., Stosick, M., Mcgindly, R.: The Automated Measurement of Software Quality. IEEE (1981)
12. Evans, M.W., Marciniak, J.J.: Software Quality Assurance and Management. John Wiley & Sons (1987)
13. Object Management Group: Unified Modeling Langauge (UML). Ver. 1.5, Formal Specification of the Object Managed Group Inc. (2003)
14. Lee, K.S., Lee, S.J.: A Weight Decision technique of Software Quality Characteristics in Software Development Life Cycle. In: Proc. of 6th ACIS ICIS '04, LA USA (2004) 338-343
15. Schulmeyer, G.C., McManus, J.I.: Handbook of Software Quality Assurance, 3rd Ed., Prentice Hall (1999)

Component Specification Model for the Web Services

Haeng-Kon Kim and Eun-Ju Park

Department of Computer Information & Communication Engineering,
Catholic University of Daegu, Kyungbuk, 712-702, South Korea
{hangkon,ejpark}@cu.ac.kr

Abstract. In our research we have created a model for automatically converting components created in different languages to web services. The components that are developed in various languages are deployed dynamically (just in time) in the web servers by using custom built deployers. Using this model the users can access the components that reside in the server using open internet standards, without having to worry about the language and platform restrictions. We then present the Interoperable Component Specification Model(ICSM) for the Web services environment.

1 Introduction

The current trend in the IT world is focused towards providing a service oriented software solutions. Service oriented architecture provides a way for developing and deploying software components as well defined services. Component based development and web services technologies provides a backbone for a true service oriented architecture. If the software components are provided as services, they will provide a high degree of reuse. The current research work focuses mainly on automatically converting components into web services, and deploying them dynamically by using custom deployers.

The proposed model highly reduces the client side execution and increases the efficiency of web servers by deploying only those components which re in demand as web services on receipt of client's request. Dynamic generation of web services from a component requires knowledge of the language and the platform of the component. The component functionalities may be converted into a web service based on the language and features of the language in which the component is implemented. Dynamic generation of web services depends on the nature of the component and the appropriate tool or technology must be used to generate a web service dynamically. We have developed custom deplorers to handle the clients request for components in different languages. The proposed model combines the best aspects of component based development and web services.

For CBD to be successful, components should be properly specified, easy to understand, sufficiently general, easy to reuse, easy to adapt, easy to maintain, easy to deploy, and easy to replace. We show the shortcomings of WSDL(Web Services Description Language) which is the standard technology used to describe Web services

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 927 – 936, 2006.

components, and propose an extension to WSDL that enhances its semantic power. And, we show that the specification model in ICSM can be easily applied in the Web services environment.

2 Related Work

Components are pieces of software that can be combined to build a bigger whole (i.e., another component, subsystem, or system). Component-based development(CBD) is an approach to software development in which software systems can be built by adapting, assembling, and wiring together existing components. The major goals of Component-based Software Engineering(CBSE) are the provision of support for the development of systems as assemblies of components, the development of components as reusable entities, and the maintenance and upgrading of systems by customizing and replacing individual components.

There are several tasks involved in CBD – specifications, development, storage, deployment, retrieval, composition, deployment, testing, and maintenance. This set of tasks defines the CBD life cycle. The details of a task are dependent upon the specific component model being followed. A component model defines how a component is specified and how components are composed. It includes the set of component types, their interfaces, and, additionally, a specification of the allowable patterns of interaction among component types.

Web services offer a platform independent solution for system integration in a distributed environment. Web services have been positioned as a key enabler to Enterprise Application Integration and B2B integration. By using XML as an on-wire standard, Web Services resolve the interoperability problem that was a problem for earlier distributed systems such as RMI, DCOM, and CORBA. Web Services are loosely coupled, self-contained, and modular software components delivered over the Internet. Web Services technology is compliant with open industry standards, like XML and HTTP. A Web services application can be easily located and invoked across the Internet. The key benefits of Web Services include lower overall integration costs, a higher degree of reusability and potential for new revenue streams.

3 Dynamic Web Service Generation

To dynamically generate a web service from a component requires knowledge on the language and the platform of the component. There are various tools and technologies available to create web services. The component functionalities may be converted into a web service based on the language and features of the language in which the component is implemented. Dynamic generation of web service depends on the nature of the component and the appropriate tool or technology must be used to generate a web service dynamically.

Figure 1 shows a logical view of the Web services architecture with the fundamental technologies involved in implementing Web Services.

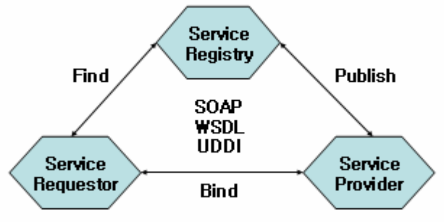


Fig. 1. Web Services Logical View

The Web services architecture is based upon the interactions between three primary roles: service provider, service registry, and service requestor. These roles interact using publish, find, and bind operations. The service provider is the business that provides access to the Web service and publishes the service description in a service registry. The service requestor finds the service description in a service registry and uses the information in the description to bind to a service. In this view of the Web services architecture, the service registry provides a centralized location for storing service descriptions. The following shows short descriptions of each technology involved in the Web services:

- XML provides the format for transferring information/data between a Web Services provider application and a Web Services client application.
- SOAP(Simple Object Access Protocol) provides standard communication channel. It is an XML-based mechanism for messaging and RPC.
- WSDL(Web Service Description Language) is a standard meta-language to describe the services offered. It is an XML equivalent of IDL.
- UDDI(Universal Description Discovery and Integration) is used for registering and locating Web Services. It provides a “Yellow page” directory of Web Services.

These technologies are emerging as Web services standards and will enable system-to-system integration that is easier than ever before. More and more web services become publicly available. They can be found in XMethods, BindingPost, and Microsoft.

WSDL describes a Web service-the services provided, where to locate them, and other supporting information. One of the advantages of WSDL is that it allows separating the abstract functionality description offered by a service form message format and communication protocol. WSDL supports both functional and non-functional descriptions. A functional description defines details on what operations are available, how and where to invoke them, and what the syntax of the resulting messages will be. Non-functional descriptions include details on policies, like security or transactions.

A web service is created for the component desired by the client. When the component is selected, then the main web service can create a dynamic web service using the component DLL. Then the address or WSDL file of this newly created web service can be given back to the client. When the client is done with the method calls, either a message can be sent to the consumer or the dynamic web service to terminate the web service. Thus the web service is created, used and then deleted when required by the client. Web services can be developed in any language and used by programs written in any language or on any platform. All the client needs is a valid WSDL file.

3.1 Steps for Converting Components into Web Services (Figure 2)

1) Client searches the registry to find the component which matches its search criteria.
2) If component is found, it contacts the service provider.
3) The service provider pulls out the component from the repository.
4) If it is a java component, it is dynamically converted into web service by using steps described in the diagram for converting java components to web services.
5) If the component is a C++ component, it is dynamically converted into web service by following steps described in the diagram for converting C++ components to web services.
6) The URI of the WSDL file is sent back to the client.
7) The client can now access the component as a web service using SOAP messages.

The current problem is to convert the already existing components into web services dynamically i.e. the components should be stored in the repository and should be converted and deployed as web services only when there is a need to use this component. The prototype of the proposed model is implemented by taking a java component and a C++ component and converting them dynamically into web services.

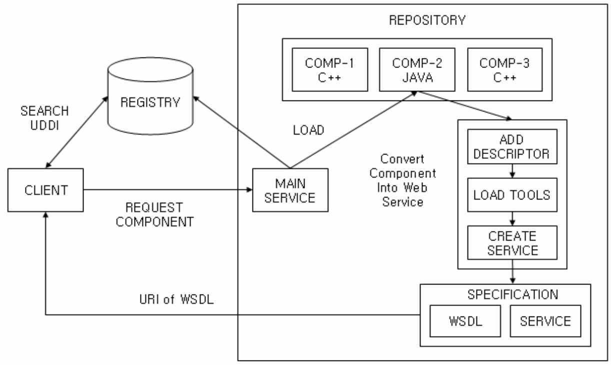


Fig. 2. Dynamic Web Service generation

4 Converting C++ Component into a Web Service

STEPS:

- The Header file of the C++ Component is edited to define the new service.
- The modified header file and the cpp file are then fed to the soapcpp2 compiler.
- The compiler generates a wsdl file and studs and skeleton for deploying that component as a web service.

The process of converting a C++ component into web service is shown in figure 3. The gSOAP toolkit is used to create web services in C++. The gSOAP Web services development toolkit offers an XML to C/C++ language binding to ease the development of SOAP/XML Web services in C and C/C++.

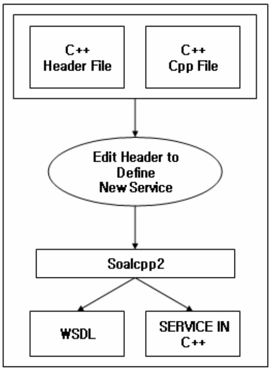


Fig. 3. Flowchart for converting C++ components to Web Service

The soapcpp2 compiler available in the gSOAP toolkit is used to convert an existing C++ component into a web service. In order to convert a C++ component into a web service, the header file the C++ file has to be edited to define the web service method operations and the data types. Now the edited header file can be compiled using soapcpp2 compiler to create a stub and skeleton for the C++ web service and a SWDL for this new web service. The step in converting a C++ component into a web service is shown in figure 4.

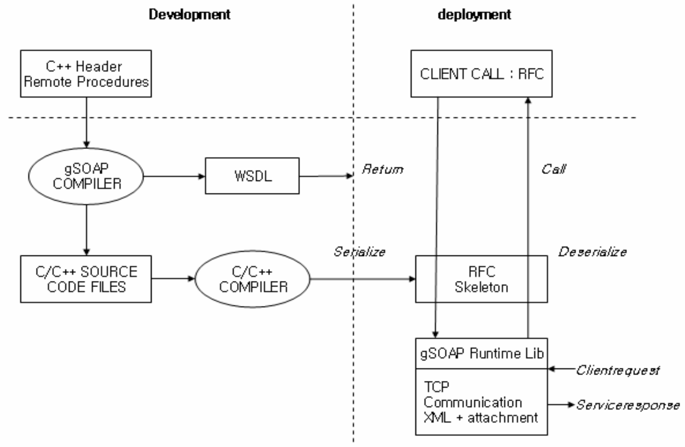


Fig. 4. Server side Development and Deployment

The Web service module was integrated into the CBD workbench, the java and the C++ components which were initially stored in the repositories were converted into web services on receipt of a client's request. When the request arrives, the main web service creates and dynamic web service using the component DLL. The RUI of the web service is then sent to the client, and the clients were able to access these components using SOAP messages. Since the components are converted into web services, they provide high interoperability, as they follow open internet standards.

The client's request for the different components were successfully served, The components are successfully converted into corresponding web services based on the request, the proposed model worked fine for the java and C++ components, and the model can also be extended to convert components of other languages to web services. The only additional step is converting that particular component into a web service, based on the client's request.

5 ICSM Component Model

Figure 5 illustrates the overall structure of our component model. A protocol is the sequence of messages that constitutes the interaction of two components. If the protocols of two components are not compatible, the components can not interoperate directly. The use of protocols to describe component interaction fosters structured, safer and potentially verifiable information exchange between components.

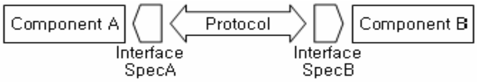
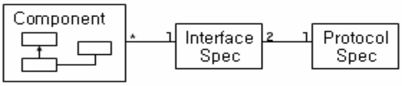


Fig. 5. Protocol Specification

In Figure 5, note that each interface specification is shown as a box with a pointed end headed toward its component. This implies that each interface specification lists only the methods provided by each component but not the methods requested from each component. As will be shown later, the protocol specification will provide an ordered list of the methods invoked by each component. In a sense, the protocol specification will work as a connector in an architectural framework. If we list methods both requested and provided in the interface specification, the assembly specification would still not have information about dependencies among the methods. Note that the protocol specification is illustrated with arrows pointing in each direction. It supports messages in both directions. The following shows the UML diagram of our component model. A component itself may exist as either in binary or in source code form, but the specification must exist in a form that users of the component can freely inspect and study before deciding to use the component.



As shown, an instance can be implemented by several different components, but only one interface per component. CORBA's CCM allows one interface per component, but COM model supports multiple interfaces per component. However, an interface in COORBA can inherit from multiple interfaces, so essentially both CCM and COM support multiple interfaces per component. In ICSM, we model a component itself as a class that has exactly one interface, but it can be implemented with a set of classes.

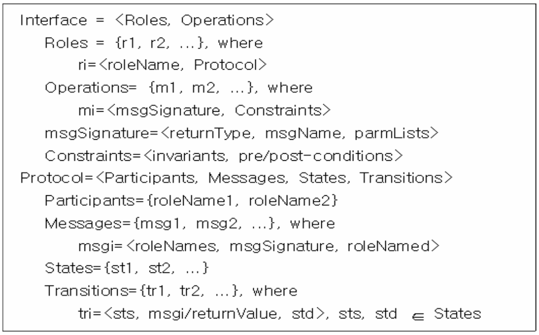
The Interface Specification lists

- full method signatures - the methods provided by the component, and
- behavior - the roles played by the component and pre/post-conditions for each method .

The Protocol Specification lists

- parties (or roles) involved in the protocol,
- messages exchanged,
- sender/receiver of the message, and
- sequence of messages sent/received.

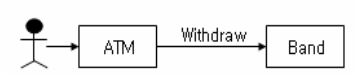
Note that the behavior specification in the interface is sometimes called a contract, and the interface is purely considered a syntactic entity. But in our model, an interface contains both syntactic and semantic entries. The following uses a formal set notation for the interface and protocol specifications.



```
Interface = <Roles, Operations>
   Roles = {r1, r2, ...}, where
       ri=<roleName, Protocol>
   Operations= {m1, m2, ...}, where
       mi=<msgSignature, Constraints>
   msgSignature=<returnType, msgName, parmLists>
   Constraints=<invariants, pre/post-conditions>
Protocol=<Participants, Messages, States, Transitions>
   Participants={roleName1, roleName2}
   Messages={msg1, msg2, ...}, where
       msgi=<roleNames, msgSignature, roleNamed>
   States={st1, st2, ...}
   Transitions={tr1, tr2, ...}, where
       tri=<sts, msgi/returnValue, std>, sts, std ∈ States
```

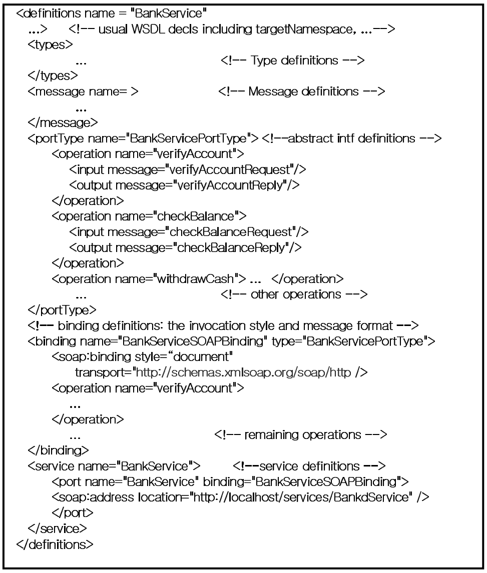
6 Behavioral and Protocol Specification in WSDL

Consider an example in the banking domain. The following shows a withdrawal attempt from a user to a Bank service via an ATM machine.



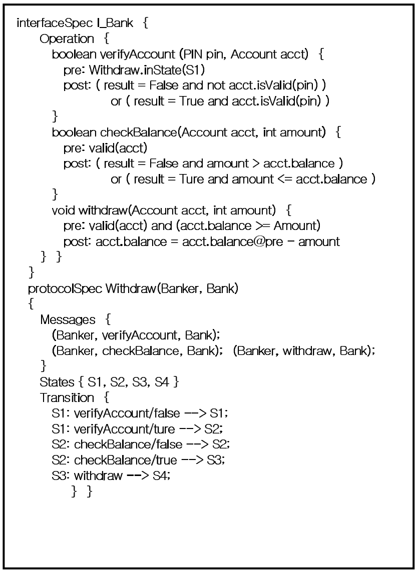
Upon a Withdraw request, the Bank service first has to verify the account, and then checks the balance of the user's account and then withdraws the cash amount from the account. The Bank service provides many banking related operations, including verifyAccount, checkBalance, and withdrawCash. The Bank service can be described as a Web service with WSDL as shown in Figure 6.

Currently, WSDL is mostly used to define the “signature” of the web service. Without support for behavioral and collaboration aspects, WSDL has the same problem as IDLs. One approach to support the behavioral specification is to provide an English-like description in WSDL for the behavioral aspect. However, this approach is subject to errors and is hard for other Web services tools to process. A better approach is to provide an extension to WSDL to effectively describe the behavior.



```
<definitions name = "BankService"
  ...>    <!-- usual WSDL decls including targetNamespace, ...-->
  <types>
       ...                         <!-- Type definitions -->
  </types>
  <message name= >                 <!-- Message definitions -->
       ...
  </message>
  <portType name="BankServicePortType"> <!--abstract intf definitions -->
     <operation name="verifyAccount">
        <input message="verifyAccountRequest"/>
        <output message="verifyAccountReply"/>
     </operation>
     <operation name="checkBalance">
        <input message="checkBalanceRequest"/>
        <output message="checkBalanceReply"/>
     </operation>
     <operation name="withdrawCash"> ...  </operation>
       ...                         <!-- other operations -->
  </portType>
  <!-- binding definitions: the invocation style and message format -->
  <binding name="BankServiceSOAPBinding" type="BankServicePortType">
     <soap:binding style="document"
         transport="http://schemas.xmlsoap.org/soap/http />
     <operation name="verifyAccount">
        ...
     </operation>
        ...                        <!-- remaining operations -->
  </binding>
  <service name="BankService">       <!--service definitions -->
     <port name="BankService" binding="BankServiceSOAPBinding">
     <soap:address location="http://localhost/services/BankdService" />
     </port>
  </service>
</definitions>
```

Fig. 6. WSDL Description of Banking Service

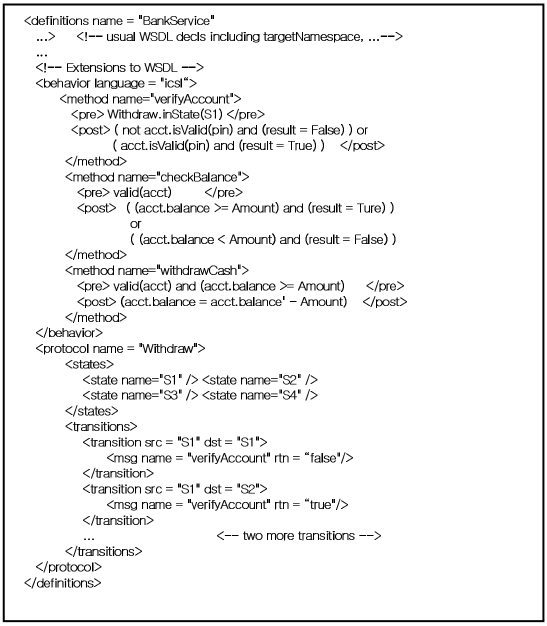


```
interfaceSpec I_Bank  {
    Operation  {
      boolean verifyAccount (PIN pin, Account acct)  {
        pre: Withdraw.inState(S1)
        post: ( result = False and not acct.isValid(pin) )
                or ( result = True and acct.isValid(pin) )
      }
      boolean checkBalance(Account acct, int amount)  {
        pre: valid(acct)
        post: ( result = False and amount > acct.balance )
                or ( result = Ture and amount <= acct.balance )
      }
      void withdraw(Account acct, int amount)  {
        pre: valid(acct) and (acct.balance >= Amount)
        post: acct.balance = acct.balance@pre - amount
    }  }
  }
  protocolSpec Withdraw(Banker, Bank)
  {
    Messages  {
      (Banker, verifyAccount, Bank);
      (Banker, checkBalance, Bank);  (Banker, withdraw, Bank);
    }
    States { S1, S2, S3, S4 }
    Transition  {
      S1: verifyAccount/false --> S1;
      S1: verifyAccount/ture --> S2;
      S2: checkBalance/false --> S2;
      S2: checkBalance/true --> S3;
      S3: withdraw --> S4;
         }  }
```

Fig. 7. Banking Service Specification

For the collaboration aspect, message sequences can be described either in BPEL (Business Process Execution Language) with Web services orchestration, or by providing an extension to WSDL with state modeling. In the next section, we show our approach to extend WSDL to support the behavioral and collaboration aspects of Web services component specification.

The interface definition can be embellished with additional behavioral and protocol specification using ICSL as shown in Figure 7. The pre and post-conditions in interfaceSpec is similar to contracts used in OCL. The Withdraw protocol lists the parties (or roles) involved in the protocol. Note that multiple components can take the role of Banker - ATM or on-line banking software. The protocol specification also shows the information of the sender/receiver of each message and the messages sequence by using the state transition model. The WSLD extension shown in Figure 8 includes behavior and protocol element. They both conform to the rules of XML syntax (or well-formed). It shows the language used for the behavioral specification is ICSL. The protocol element in WSDL helps eliminate the message sequences ordering problem. The tool that reads this specification should be able to access the ICSL tool which can interpret and process the pre and pose-conditions for each operation.



```
<definitions name = "BankService"
  ...>    <!-- usual WSDL decls including targetNamespace, ...-->
  ...
  <!-- Extensions to WSDL -->
  <behavior language = "icsl“>
      <method name="verifyAccount">
        <pre> Withdraw.inState(S1) </pre>
        <post> ( not acct.isValid(pin) and (result = False) ) or
               ( acct.isValid(pin) and (result = True) )   </post>
       </method>
       <method name="checkBalance">
         <pre> valid(acct)      </pre>
         <post>  ( (acct.balance >= Amount) and (result = Ture) )
                  or
                  ( (acct.balance < Amount) and (result = False) )
       </method>
       <method name="withdrawCash">
         <pre> valid(acct) and (acct.balance >= Amount)     </pre>
         <post> (acct.balance = acct.balance' – Amount)    </post>
       </method>
  </behavior>
  <protocol name = "Withdraw">
       <states>
          <state name="S1" /> <state name="S2" />
          <state name="S3" /> <state name="S4" />
       </states>
       <transitions>
          <transition src = "S1" dst = "S1">
               <msg name = "verifyAccount" rtn = “false"/>
          </transition>
          <transition src = "S1" dst = "S2">
               <msg name = "verifyAccount" rtn = “true"/>
          </transition>
          ...                         <-- two more transitions -->
       </transitions>
  </protocol>
</definitions>
```

Fig. 8. Extended WSDL Specification

7 Conclusions and Future Work

In this paper, we have created a model for automatically converting components created in different languages to web services. We have motivated and laid the groundwork for the integration of the Web services and Component Based Development. Web services represent a evolution of the Web to allow applications to interact over the Internet in an open and flexible way. Interoperability between different systems is one of the primary reasons for using Web Services. Web Services are gong to play an important role in the future of distributed computing, significant impacting application and system development.

Also, to address that one component can interact with another, we presented ICSM and its support of behavioral and collaboration aspect among components. The prominent use of XML and the need for integrating business enterprise applications have led to the success of Web services. One of the major thrusts for web services has been the promise of easy application integration by creating a WSDL representation for any applications (including legacy), which can then be registered to a central registry for later use by a thirdparty looking for an application.

The major contribution of this paper is an extension of the Web services component model to support semantics lacking in WSDL. The current Web services tools would not know how to deal with the extension. A proper mechanism is needed to tell the Web services tool to refer to a proper tool to process the semantic extension. We are working on refining the extension semantics and investigating a proper way of integrating the processing the semantic information under the Axis Web services tool.

Currently, the WSDL extension for the behavior and protocol elements is manually generated. We are considering tool support to automatically generate the WSDL codes from the specification written in ISDL.

References

1. G. Heineman and W. Councill. Component-based Software Engineering, Putting the Pieces Together, Addision-Wesley, 2001
2. D. Garlan, R. Monroe, and D. Wile, Acme : Architectural Description of Component-Based Systems, Foundations of Component-Based Systems, Gary T. Leavens and Murali Sitaraman (eds), Cambridge University Press, pp. 47–68, 2000.
3. I. Cho and J.D. McGregor, A Formal Approach to Specifying and Testing the Interoperation between Components, 38th Annual ACM Southeast Conference, 2000.
4. Web Services Description Language(WSDL) 1.1 W3C Note, World Wide Web Consortium, http://www.w3.org/TR/wsdl, March 2001.
5. Gottschalk K, Graham S, Kreger H, Snell J, Introduction to Web Services Architecture, IBM Systems Journal, Vol. 41, No. 2, 2002.

A Data-Driven Approach to Constructing an Ontological Concept Hierarchy Based on the Formal Concept Analysis

Suk-Hyung Hwang[1], Hong-Gee Kim[2], Myeng-Ki Kim[2],
Sung-Hee Choi[3], and Hae-Sool Yang[4]

[1] Digital Enterprise Research Institute, Seoul National University,
Division of Computer and Information Science, SunMoon University,
100 Kal-San-Ri, Tang-Jeong-Myon, A-San, Chung-Nam, 336-840 Korea
shwang@sunmoon.ac.kr
[2] Digital Enterprise Research Institute, Seoul National University,
28-22 Yeonkun-Dong, Chongno-Ku, Seoul 110-749, Korea
{hgkim, meeree}@snu.ac.kr
[3] Division of Computer and Information Science, SunMoon University,
100 Kal-San-Ri, Tang-Jeong-Myon, A-San, Chung-Nam, 336-840 Korea
shchoi@sunmoon.ac.kr
[4] Graduate School of Venture, Hoseo University,
29-1 Se-Chul-Ri, Bae-Bang-Myon, A-San, Chung-Nam, 336-795 Korea
hsyang@office.hoseo.ac.kr

Abstract. An ontology is a formal, explicit specification of a domain. An important benefit of using an ontology during software development is that it enables the developer to reuse and share application domain knowledge using a common vocabulary across heterogeneous software platforms and programming languages. One of the most important components of ontologies is concept hierarchy, which models the information on the domain of interest in terms of concepts and subsumption relationships between them. However, it is extremely difficult and time-consuming for human experts to discover concepts and construct concept hierarchies from the domain.

In this paper we introduce Formal Concept Analysis(FCA) as the basis for a practical and well founded methodological approach to the construction of concept hierarchy. We present a semi-automatic tool, FCAWIZARD, to support the concept hierarchy construction. Based on the FCAWIZARD, we are now exploring a data-driven approach to construct medical ontologies from some medical data contained in clinical documents. We discuss the basic ideas of our work and its current state as well as the problems encountered and future directions.

1 Introduction

An ontology is an explicit specification of a conceptualization [1]. One of the most important components of ontologies are concept hierarchies, which model the information on the domain of interest in terms of concepts and subsumption

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 937–946, 2006.

relationships between them. In object oriented systems this hierarchy is referred to as *class hierarchy* that is the cornerstone of inheritance, subtype polymorphism and software reuse, etc [2]. In knowledge-based systems research, this hierarchy is often termed an *ontology* that has many applications in many areas including natural language translation, medicine, standardization of product knowledge, electronic commerce, and semantic webs [3]. Recently, this concept hierarchy or ontology information can be modelled in UML class diagrams and Object Constraint Language(OCL) constraints [4, 5]. Software design artifacts, such as UML class diagrams, and even source code can be generated directly from ontologies, thus effectively speeds up the software development process. Moreover, in the early phases of software analysis and modeling, ontologies may lay the ground for better integrated application systems and for better reusable models and application components [6].

An ontology is based on the concept hierarchy to represent the concept generalization structure generated from the underlying data. Concept hierarchies play an important role as the backbone of object-oriented systems as well as some knowledge-based systems, etc. In each of these systems, the developer attempts to capture the main concepts and concept hierarchies of the domain relevant to that system. However, manual construction of concept hierarchies is a complex and time-consuming process. It is extremely difficult for human experts to discover concepts and their generalization structures from given data. One solution to the problem is to build concept hierarchy automatically or at least, use semi-automatic methods based on the formal concept analysis.

Our aim in this research is to adopt the FCA as a framework for the identifying concepts and the construction of ontological concept hierarchies. More specifically we shall present a novel and data-driven approach to the semi-automatic construction of ontological concept hierarchies from given data of a domain of interest. The rest of the paper is organized as follows: in Section 2, we introduce some basic notions of the formal concept analysis. In Section 3, it is presented not only an overview of construction of ontological concept hierarchy, but also a semi-automatic tool, "FCAwizard", for concept hierarchy construction. Lastly, we conclude the paper, giving the current state, the problems encountered and future directions of our research in Section 4.

2 Basic Notions

Formal Concept Analysis(FCA) is a method mainly used for the analysis of data, i.e. for investigating and processing explicitly given information. Such data are structured into units which are formal abstractions of concepts of human thought allowing meaningful comprehensible interpretation. FCA was introduced as a mathematical theory modeling the concept 'concept' in terms of lattice theory [7, 8]. This approach arose independently of ontologies, resulting in a different formalization of concepts.

2.1 Contexts and Concepts

FCA starts with a *formal context* that is comprised of a set of objects, a set of attributes and a relation describing which objects possess which attributes. In the formal definition, the set of objects is denoted by $\mathcal{O}$, and the set of attributes is denoted by $\mathcal{A}$.

Definition 1. *A* ***formal context*** *is a triple* $(\mathcal{O}, \mathcal{A}, \mathcal{R})$*, where* $\mathcal{O}$ *is a set of objects and* $\mathcal{A}$ *is a set of attributes, and* $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{A}$ *is a binary relation between* $\mathcal{O}$ *and* $\mathcal{A}$*. In order to express that an object* o *is in a relation with an attribute* a*, we write* $(o, a) \in \mathcal{R}$ *and read it as "the object* o *has the attribute* a*".*

Table 1 shows a formal context of some animals that is based on the set of objects $\mathcal{O}$ and the set of their attributes $\mathcal{A}$ as follows: $\mathcal{O} = \{\textit{sea-turtle}, \textit{dog}, \textit{cat}, \textit{cow}\}$, $\mathcal{A} = \{\textit{four-legged}, \textit{land-living}, \textit{water-living}, \textit{livestock}, \textit{pet}\}$ and the incidence relation $\mathcal{R}$ is given by the cross table. Let $(\mathcal{O}, \mathcal{A}, \mathcal{R})$ be a context. For $O \subseteq \mathcal{O}$, we define $\mathsf{intent}(O) := \{a \in \mathcal{A} | \forall o \in O : (o, a) \in \mathcal{R}\}$, and, dually for $A \subseteq \mathcal{A}$, we define $\mathsf{extent}(A) := \{o \in \mathcal{O} | \forall a \in A : (o, a) \in \mathcal{R}\}$. The function intent maps a set of objects into the set of attributes common to the objects in $O(\mathsf{intent} : 2^O \rightarrow 2^A)$, whereas extent is the dual for attributes sets($\mathsf{extent} : 2^A \rightarrow 2^O$). These two functions form a *Galois connection* between the objects and attributes of the context.

Table 1. Formal context for animals

	four-legged	*land-living*	*water-living*	*livestock*	*pet*
sea-turtle	×		×		
dog	×	×			×
cat	×	×			×
cow	×	×		×	

The central notion of FCA is the *(formal) concept*. Objects from a context share a set of common attributes and vice versa. Concepts are pairs of objects and attributes which are synonymous and thus characterize each other. Concepts can be imagined as maximal rectangles in the context table. If we ignore the sequence of the rows and columns we can identify even more concepts. The formal definition of concept is given in the following:

Definition 2. *Let* $(\mathcal{O}, \mathcal{A}, \mathcal{R})$ *be a context. A* ***formal concept*** *is a pair* (O, A) *with* $O \subseteq \mathcal{O}$ *is called* ***extension****,* $A \subseteq \mathcal{A}$ *is called* ***intension****, and*

$$(O = \mathsf{extent}(A)) \wedge (A = \mathsf{intent}(O)).$$

In other words a concept is a pair consisting of a set of objects and a set of attributes which are mapped into each other by the Galois connection. The set of all concepts of the context $\mathcal{C} = (\mathcal{O}, \mathcal{A}, \mathcal{R})$ is denoted by $B(\mathcal{C})$ or $B(\mathcal{O}, \mathcal{A}, \mathcal{R})$, i.e., $B(\mathcal{C}) = \{(O, A) \in 2^{\mathcal{O}} \times 2^{\mathcal{A}} | O = \mathsf{extent}(A) \wedge A = \mathsf{intent}(O)\}$. For example, all concepts of the context of Table 1 are as follows:

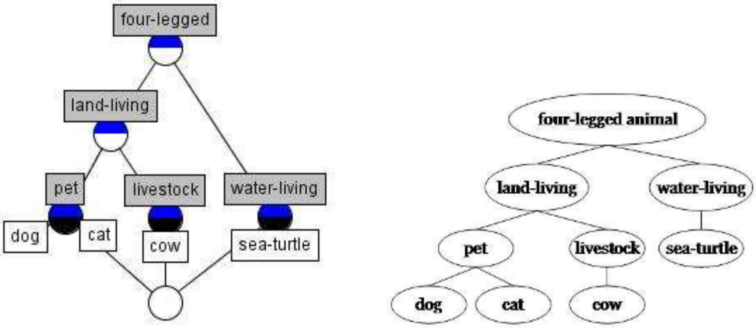


Fig. 1. Concept lattice and the corresponding ontological concept hierarchy for the context of Table 1

$(\{sea\text{-}turtle, dog, cat, cow\}, \{four\text{-}legged\}), (\{dog, cat, cow\}, \{four\text{-}legged, land\text{-}living\}), (\{dog, cat\}, \{four\text{-}legged, land\text{-}living, pet\}), (\{cow\}, \{four\text{-}legged, land\text{-}living, livestock\}), (\{sea\text{-}turtle\}, \{four\text{-}legged, water\text{-}living\}), (\emptyset, \{four\text{-}legged, land\text{-}living, water\text{-}living, livestock, pet\})$.

The set of formal concepts is organized the partial ordering relation $\leq$ -to be read as "is a subconcept of"- as follows:

Definition 3. *For a Formal Context* $\mathcal{C} = (\mathcal{O}, \mathcal{A}, \mathcal{R})$ *and two Concepts* $c_1 = (O_1, A_1)$, $c_2 = (O_2, A_2) \in B(\mathcal{C})$ *the* ***subconcept-superconcept*** *relation is given by* $(O_1, A_1) \leq (O_2, A_2) \Leftrightarrow O_1 \subseteq O_2 (\Leftrightarrow A_1 \supseteq A_2)$.

This relationship shows that the dualism exists between attributes and objects of concepts. A concept $c_1 = (O_1, A_1)$ is a subconcept of concept $c_2 = (O_2, A_2)$ iff the set of its objects is a subset of the objects of c_2. Or an equivalent expression is iff the set of its attributes is a superset of the attributes of c_2. That is, a subconcept contains fewer objects and more attributes than its superconcept. The set of all formal concepts of a context $\mathcal{C}$ with the *subconcept-superconcept realtion* $\leq$ is always a complete lattice[1], called the *(formal) concept lattice* of $\mathcal{C}$ and denoted by $\mathcal{L} := (B(\mathcal{C}), \leq)$. Figure 1 shows the concept lattice and the corresponding ontological concept hierarchy for the *Animal* context of Table 1.

2.2 Many-Valued Contexts

FCA may be applied to data in which objects are interpreted as having attributes with values. That is, in real world, the attribute is not only a property that an object may have nor not have. Attributes can have values. For example, the "color" attribute may have many values such as "yellow", "green" or "red". We call them *many-valued attributes*, in contrast to the *one-valued attributes* considered so far. In this case the basic data is stored in a *many-valued context.*

[1] That is, for each set of formal concepts, there exists always a unique greatest common subconcept and a unique least common superconcept.

Definition 4. *A **many-valued context** $(\mathcal{O}, \mathcal{A}, \mathcal{V}, \mathcal{R})$ is a set of objects $\mathcal{O}$, a set of attributes $\mathcal{A}$, a set of possible values $\mathcal{V}$, and ternarry relation $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{A} \times \mathcal{V}$, where the following holds: $(o, a, v_1) \in \mathcal{R} \wedge (o, a, v_2) \in \mathcal{R} \Rightarrow (v_1 = v_2)$.*

$(o, a, v) \in \mathcal{R}$ indicates that the object o has a value v for the attributes a. Every attribute $a \in \mathcal{A}$ can be regarded as a partial function $a : \mathcal{O} \rightarrow \mathcal{V}$, dually every object $o \in \mathcal{O}$ can be regarded as a partial function $o : \mathcal{A} \rightarrow \mathcal{V}$. A many-valued context is called *complete* if every attribute $a \in \mathcal{A}$ is a function(equivalently, every object $o \in \mathcal{O}$ is a function). Like the one-valued context, many-valued context can be represented as a matrix, the rows of which are labeled by the objects and the columns of which are labeled by the attributes. However the entry in row o and column a has attribute value $a(o)$. Table 2 shows an example of a many-valued context for some fruits.

Table 2. An example of many-valued context

	kind	color	habitat	price
fruit 1	apple	yellow	Korea	\$1.15
fruit 2	grapefruit	yellow	France	\$7.25
fruit 3	kiwi	green	New Zealand	\$4.70
fruit 4	apple	red	Korea	\$2.15

In order to get concepts out of this many-valued context and draw the concept lattice, we have to transform the many-valued context into a one-valued context according to certain rules. The new one-valued context is called the *derived context.* The concepts of derived one-valued context are interpreted as the concepts of the many-valued context. The process of transformation is called *conceptual scaling.* And this process is not uniquely determined, but depends on the transformation rules. Formally, a many-valued context is transformed by constructing a *scaling* for each attribute. The scales are used to construct one-valued contexts for each attribute which are then combined or joined to form a one-valued context which represents the original many-valued context.

Definition 5. *A **scale** for an attribute $a \in \mathcal{A}$ from a many-valued context $(\mathcal{O}, \mathcal{A}, \mathcal{V}, \mathcal{R})$ is a one-valued context $S_a = (\mathcal{O}_a, \mathcal{A}_a, \mathcal{R}_a)$ with $\mathcal{A}_a \subseteq \mathcal{V}$ is a set of values of the attribute $a \in \mathcal{A}$ and $\mathcal{O}_a = \{v \in \mathcal{V} | (o, a, v) \in \mathcal{R}\} \subseteq \mathcal{A}_a$.*

The simplest version of scaling is called *plain scaling.* In the case of plain scaling the derived one-valued context is obtained from the many-valued context $(\mathcal{O}, \mathcal{A}, \mathcal{V}, \mathcal{R})$ and the scale context $S_a, a \in \mathcal{A}$ as follows: The object set $\mathcal{O}$ remains unchanged, every many-valued attribute a is replaced by the scale attribute of the scale S_a.

Definition 6. *Let $\mathbb{K} = (\mathcal{O}, \mathcal{A}, \mathcal{V}, \mathcal{R})$ be a complete many-valued context. Then $\mathbb{K}^n = (\mathcal{O}, \mathcal{N}, \mathcal{J})$ is called **drived context via plain scaling** of $\mathbb{K}$ if*

$$\mathcal{N} = \{(a, v) \in \mathcal{A} \times \mathcal{V} | \exists o \in \mathcal{O} : a(o) = v\}, \mathcal{J} = \{(o, (a, v)) \in \mathcal{O} \times \mathcal{N} | (o, a, v) \in \mathcal{R}\}.$$

Table 3. Some examples of Scale contexts

S_{color}	*yellow*	*green*	*red*
yellow	×		
green		×	
red			×

S_{price}	*cheap*	*mid-range*	*expensive*
$0 \leq$ price < 2	×		
$2 \leq$ price < 4		×	
$4 \leq$ price < 6		×	
$6 \leq$ price < 8			×

Table 4. Derived context $\mathcal{C}_{color}$ and $\mathcal{C}_{price}$

$\mathcal{C}_{color}$	*yellow*	*green*	*red*
fruit 1	×		
fruit 2	×		
fruit 3		×	
fruit 4			×

$\mathcal{C}_{price}$	*cheap*	*mid-range*	*expensive*
fruit 1	×		
fruit 2			×
fruit 3		×	
fruit 4		×	

If we imagine a many-valued context as represented by a matrix, we can visualize plain scaling as follows: Every attribute value $a(o)$ is replaced by the row of the scale context S_a which belongs to $a(o)$.

To make this situation clearer consider a many-valued context in table 2 and take scale contexts(S_{color} and S_{price} in Table 3) for the attributes *color* and *price*, respectively. Now we can do plain scaling with the scale contexts. As shown in Table 4, we can derive one-valued contexts $\mathcal{C}_{color}$ and $\mathcal{C}_{price}$ from the original many-valued context according the scales.

When transforming a many-valued context $(\mathcal{O}, \mathcal{A}, \mathcal{V}, \mathcal{R})$, there will be a one-valued context for each attribute of $\mathcal{A}$. Therefore, if $|\mathcal{A}| \geq 1$, the multiple contexts need to be combined to form one unified context. *Apposition* is the operation that combine multiple contexts having the same set of objects in common.

Definition 7. *Let* $\mathcal{C}_1 = (\mathcal{O}, \mathcal{A}_1, \mathcal{R}_1)$ *and* $\mathcal{C}_2 = (\mathcal{O}, \mathcal{A}_2, \mathcal{R}_2)$ *be contexts. The* ***apposition*** *of* $\mathcal{C}_1$ *and* $\mathcal{C}_2$ *is defined as* $\mathcal{C}_1|\mathcal{C}_2 := (\mathcal{O}, \dot{\mathcal{A}}_1 \cup \dot{\mathcal{A}}_2, \dot{\mathcal{R}}_1 \cup \dot{\mathcal{R}}_2)$, *where,* $\dot{\mathcal{A}}_j := \{j\} \times \mathcal{A}_j$ *and* $\dot{\mathcal{R}}_j := \{j\} \times \mathcal{R}_j$ *for* $j \in \{1, 2\}$.

Table 5. Context table for the apposition of derived contexts $\mathcal{C}_{kind}|\mathcal{C}_{color}|\mathcal{C}_{habitat}|\mathcal{C}_{price}$.(where, *Ap*=Apple, *Gf*=GrapeFruit, *Kw*=Kiwi, *Ye*=Yellow, *Gr*=Green, *Re*=Red, *Kr*=Korea, *Fr*=France, *Nz*=New Zealand, *Ch*=Cheap, *Mr*=Mid-range, *Ex*=Expensive).

	Ap	*Gf*	*Kw*	*Ye*	*Gr*	*Re*	*Kr*	*Fr*	*Nz*	*Ch*	*Mr*	*Ex*
fruit 1	×			×			×			×		
fruit 2		×		×				×				×
fruit 3			×		×				×		×	
fruit 4	×					×	×				×	

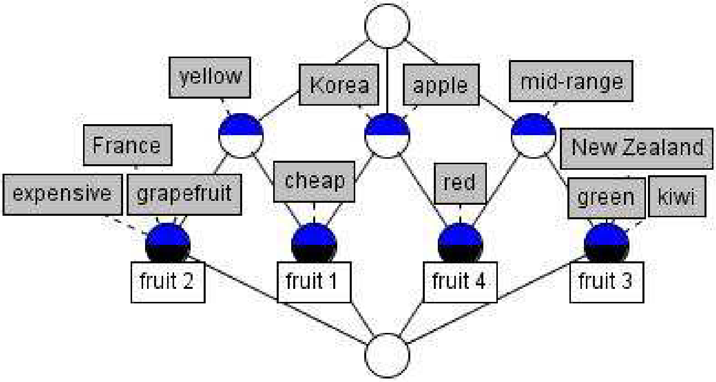


Fig. 2. Concept lattice for the context of Table 5

Table 5 shows a context table as a result of the apposition of derived contexts that can be derived from scale contexts and the many-valued context of table 2. Figure 2 shows the concept lattice generated from the many-valued context.

3 Building Ontological Concept Hierarchy

3.1 Concept Lattice Construction

A concept lattice (a concept hierarchy) can be represented graphically using line (or Hasse) diagrams. These structures are composed of nodes and links. Each node represents a concept with its associated intensional description. The links connecting nodes represent the subconcept/superconcept relation among them. This relation indicates that the parent's extension is a superset of each child's extension. Attributes propagate along the edges to the bottom of the diagram and dually objects propagate to the top of the diagram. More abstract or general nodes occur higher in the hierarchy, whereas more specific ones occur at lower levels. Now, we can summarize the above considerations as an algorithm to construct concept lattice in **Algorithm 1:** GenerateConcepts and BuildConceptLattice.

Based on the definitions and algorithm, we implement a semi-automatic tool, FCAWIZARD, to support both building concept lattice and scaling of many-valued context for the ontological concept hierarchy construction. FCAWIZARD creates concept lattices in a number of steps(see figure 3): first it builds one-valued context(in case of many-valued context, some scalings and appositions should be done); from this it creates a list of formal concepts. They are then put into the algorithm to build the concept lattice, and displayed to a line diagram(Hasse diagram) format. Now given a concept lattice $\mathcal{L} := (B(\mathcal{C}), \leq)$ of a formal context $\mathcal{C} = (\mathcal{O}, \mathcal{A}, \mathcal{R})$, we transform it into the corresponding ontological concept hierarchy (N, E) as follows:

Algorithm 1. GenerateConcepts and BuildConceptLattice

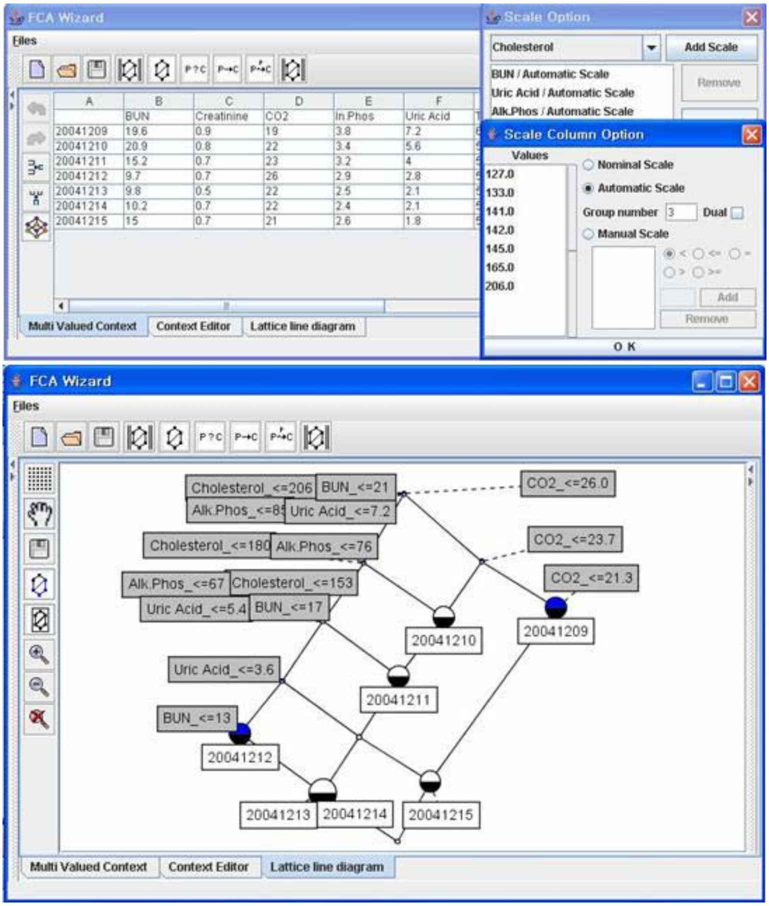
```
// INPUT : a formal context $\mathcal{C} := (\mathcal{O}, \mathcal{A}, \mathcal{R})$
// OUTPUT : Concept Lattice $\mathcal{L} := (B(\mathcal{C}), E_{\leq})$
for all $o \in \mathcal{O}$ do
  $B(\mathcal{C}) \leftarrow B(\mathcal{C}) \cup (\mathsf{extent}(\mathsf{intent}(o)), \mathsf{intent}(o))$;
end for
for all $c \in B(\mathcal{C})$ do
  for all $o \in (\mathcal{O} - \mathsf{extent}(c))$ do
    $X \leftarrow \mathsf{extent}(c) \cup \{o\}$;
    if $(\mathsf{extent}(\mathsf{intent}(X)), \mathsf{intent}(X)) \notin B(\mathcal{C})$ then
      $B(\mathcal{C}) \leftarrow B(\mathcal{C}) \cup (\mathsf{extent}(\mathsf{intent}(X)), \mathsf{intent}(X))$;
    end if
  end for
end for
for all $c_1 \in B(\mathcal{C})$ do
  for all $c_2 \in B(\mathcal{C}) - \{c_1\}$ do
    if $(c_1 \leq c_2) \wedge (\nexists c_3 \in B(\mathcal{C}) - \{c_1, c_2\}[(c_1 \leq c_3) \wedge (c_3 \leq c_2)])$ then
      $E_{\leq} \leftarrow E_{\leq} \cup \{(c_1, c_2)\}$;
    end if
  end for
end for
```

$$\begin{aligned} N &= \mathcal{O} \cup \{Y | (X, Y) \in B(\mathcal{C})\}, \\ E &= \{(o, Y_1) | (\mathsf{extent}(\mathsf{intent}(o)), \mathsf{intent}(o)) = (X_1, Y_1)\} \\ &\qquad \cup \{(Y_1, Y_2) | (X_1, Y_1) \leq (X_2, Y_2)\} \end{aligned}$$

3.2 FCAwizard and Its Applications

Our new tool, FCAwizard is based on ConExp [9] which does not have any scaling features for the many-valued context, but is easy to use and has powerful visualization system. We extend ConExp by implementing some additional functions for the editing of many-valued context and the scaling procedures. FCAwizard provides three scaling approaches: (1) Nominal scale is simple and plain scaling approach and can be used by all kinds of values. (2) Manual scale is very powerful scaling approach that can express scaling methods in every detail. Since it gives order to string values, it can process string values in the manner of numeric values. (3) Automatic scale is the most useful scale approach. It divides values automatically and then, applies ordinal or interordinal scaling. Figure 3 shows some screenshots of FCAwizard.

FCAwizard is successfully applied to analyze some medical domain data. We show some applications of FCAwizard to verify the potentiality and usability of our tool in medical domain. We analyze clinical chemistry test result of one patient as shown in Figure 3. It examines the urine specimen. By using FCAwizard, this context can be analyzed and scaled several ways. However, we mainly inspect the change of daily health state, hence checkup dates are used as objects and each test results are used as attributes. It forms many-valued context that



A	B	C	D	E	F
	BUN	Creatinine	CO2	In.Phos	Uric Acid
20041209	19.6	0.9	19	3.8	7.2
20041210	20.9	0.8	22	3.4	5.6
20041211	15.2	0.7	23	3.2	4
20041212	9.7	0.7	26	2.9	2.8
20041213	9.8	0.5	22	2.5	2.1
20041214	10.2	0.7	22	2.4	2.1
20041215	15	0.7	21	2.6	1.8

Cholesterol

BUN / Automatic Scale
Uric Acid / Automatic Scale
Alk.Phos / Automatic Scale

Values
127.0
133.0
141.0
142.0
145.0
165.0
206.0

○ Nominal Scale
● Automatic Scale
Group number 3 Dual □
○ Manual Scale
● < ○ <= ○ =
○ > ○ >=

Fig. 3. Screenshots of FCAwizard

has a numeric values. Figure 3 shows the line diagram and interesting results. It is quite readable and can be easily checked daily changes of patient records based on the concept lattice(concept hierarchy).

4 Conclusions

In this article, we present our research-in-progress concerning FCAwizard, the semi-automatic tool for ontological concept hierarchy construction. Our work is based on the concept lattice of the Formal Concept Analysis which allows to construct a "well defined" ontological concept hierarchy with maximally factorized properties. Within the framework of Formal Concept Analysis, ontologies can be considered(under some constraints) as (many-valued) contexts. Therefore, FCA can help structure and build ontologies. This is because FCA can express ontology in a lattice. The lattice is easy for people to understand and can serve as a guideline for ontology building.

As an application of FCAWIZARD, we present some results of clinical chemistry test. From the result, we show its usefulness and potentiality in medical domain. FCAWIZARD is a helpful tool for medical domain users to analyze various medical data without mathematical understanding of Formal Concept Analysis. So, it can be also applied various medical data and other related domain data. FCAWIZARD is still being developed, but we have a plan to collect feedbacks and build concept hierarchies by analyzing various medical data using FCAWIZARD. From the concept hierarchies, we can acquire well-structured facts and knowledges in medical domain. It would be helpful for building of various ontologies in medical domain.

Acknowledgements

This research was supported by a grant of the Interoperable EHR Research and Development Center(A050909), Ministry of Health & Welfare, Republic of Korea.

References

1. T. R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. Presented at the Padua workshop on Formal Ontology, March 1993.
2. James Odell. Advanced Object-Oriented Analysis and Design. Cambridge University Press. 1998.
3. N. Guarino. Formal Ontology and Information Systems. in Proc. FOIS'98, Trento, Italy, July 1998.
4. Cranefield, S., and Purvis, M. UML as an Obtology Modeling Language. Proc. of the Workshop on Intelligent Information Integration, 16th Int. Joint Conference on AI(IJCAI-99). Germany, 1999.
5. Cranefield, S. UML and the Semantic Web. Proc. of the International Semantic Web Working Symposium. Palo Alto, 2001.
6. W. Hesse. Ontologies in the Software Engineering process. in: R. Lenz et al. (Hrsg.): EAI 2005 - Tagungsband Workshop on Enterprise Application Integration, GITO-Verlag Berlin 2005.
7. Garrett Birkhoff. Lattice Theory. American Mathematical Society Coll. Pub. 25, 1940.
8. B. Ganter and R. Wille, "Formal Concept Analysis, Mathematical Foundations," Springer-Verlag, 1999.
9. Serhiy A. Yevtushenko, System of data analysis "Concept Explorer".(In Russian). Proceedings of the 7th national conference on Artificial Intelligence KII-2000, pp. 127–134, Russia, 2000.

Web-Document Filtering Using Concept Graph*

Malrey Lee[1], Eun-Kwan Kang[2], and Thomas M. Gatton[3]

[1] School of Electronics & Information Engineering , Chonbuk National University,
664-14, DeokJin-dong, JeonJu, ChonBuk,, Korea, 561-756
Fax.: 82-63-270-3403
mrlee@chonbuk.ac.kr
[2] Depart of Multimedia Engineering, JeonJu University, Korea
[3] School of Engineering and Technology, National University, 11255 North Torrey Pines Road,
La Jolla, CA 92037, USA

Abstract. This paper introduces a retrieval method based on conceptual graph. A hyperlink information is essential to construct conceptual graph. The information is very useful as it provides summary and further linkage to construct conceptual graph that has been provided by human. It also has a property which shows review, relation, hierarchy, generality, and visibility. Using this property, we extracted the keywords of web documents and made up of the conceptual graph among the keywords sampled from web pages. This paper extracts the keywords of web pages using anchor text one out of hyperlink information and makes hyperlink of web pages abstract as the link relation between keywords of each web page. I suggest this useful retrieval method providing querying word extension or domain knowledge by conceptual graph of keywords.

1 Introduction

At present, a retrieval engine is essential to search for the information in the internet. However, it does not present a satisfactory solution to user's desire for the information. Most of retrieval engines store keywords of web documents, which are indexed. Retrieval results differ according to the selection and combination of words, because they are presented through matching to user's queried words. User makes efforts of reading many web documents to find out one which can satisfy his/her desire for the information among listed web documents.

The most frequent words are considered as keywords in keyword extracting method of web documents[1]. Keyword extracting method through context analysis has time and technical limit to have to process a natural language. The list of web documents which have only user's queried words and keywords in a retrieval result has a limit listing unnecessary web documents due to the words of the same, redundant, or various meaning. A retrieval method based on concept has been researched to provide concerned documents or domain knowledge using concept in order to complement the limit [2, 3, 4].

* This paper was supported by research funds of ChonBuk National University.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 947 – 954, 2006.

Hyperlink is drawn up by author of web document, and it is composed of anchor text and link to referred web document. When we move to another web document from an web one, we refer to anchor text. Anchor text is recognized implicitly as the representation of the contents of web document connected by hyperlink. Keyword extracting method using an explanation described by author is more accurate and effective than the extraction through context analysis. Anchor text is a good information for web document. Existing anchor text in web document, however, was not connected to the web one of hyperlink, but to the web one of anchor text[5].

Hyperlink information has used only literal meaning, and it has overlooked the relationship between web documents[6,7].

In this paper, keywords of web documents can be extracted quickly using anchor text of hyperlink information. Hyperlink between web documents is abstracted into a link between keywords of each web one. We present a method which makes possible concept-based retrieval providing query extension or domain knowledge by construction of concept graph of keywords.

2 Concept-Based Retrieval

Concept is the process of expanding a particular word to a similar word or a relational one. Concept-based retrieval is the method which makes possible similar, redundant, and hierarchical representation of word with retrieval expansion using conceptual relationship of words by analyzing word meaning without relying on its spelling only. The method is similar to human way of thought, and it is more effective than other retrieval one. Existing retrieval methods list concerned web documents through simple matching queried words to words in web documents. In the method, the selection and combination of words is important, because the listed contents are different according to the selection and combination of words. In case of simple matching, no classification of similar and redundant words increases amount of listed documents.

Concept extracting method is divided into a classification of documents in a predefined concept, and the clustering with automatic generation of concept without predefined concept.

INQUERY and EXCITE are the system to classify document using predefined concept. INQUERY system uses the method to map concept to document using modified bayesian network, after it conceptualizes predefined terminology dictionary using LEX[8]. EXCITE system uses easily extensible “Intelligent Concept Extraction” using modified "Latent Semantic Indexing” without counting on computational ability of computer [4]. The system which generates and extracts a concept automatically without predefined concepts draws up and constructs thesaurus and concept respectively based on the contents of documents using genetic algorithm and self-organized network [5]. The method which is extensible makes an automatic classification, but it requires so much time for initial work. In addition to the method, standardization of IETF is proceeding for concept-extracting work in retrieval system based on additional entry of information for concept extraction by web document author with meta tag added to HTML [6]. The method is so fast in retrieval engine without the necessity of concept-extracting work, and it does not degrade the performance even with increased storage of documents. In case of web documents without the information, however, other methods should be used. Concept is used for retrieval expanding inputted query or using various methods to represent concept.

Query informs the system of the fact user intends to retrieve. There are queries of extensive subject, details, and similar documents[8]. Extensive subject query means that it is searched for easily from many documents full of the concerned facts. For example, documents resulting from query like "search for the information about web browser" can be represented in various words. Automatic and effective method, however, should be provided, because many concerned documents make accurate results difficult. Detail query is the most of queries made to a retrieval engine. Necessary information of it is contained in a few documents only. If the words are changed, the result is difficult to be searched for. Query of similar documents produces the result by measuring the similarity of document. The query requires natural language process rather than simple word matching. The weakness of details query is that a single query cannot deal with tremendous amount of data processed by each retrieval system. With a single query, threshold of the result is much higher than one of the result determined by the system. Therefore, it should have multiple queries, not a single query. In order to make multiple ones, we should have basic knowledge or clear understanding about the things we want to know.

In this paper, we present a more effective method in processing extension of details query and wide-ranged subject query. Extracted keywords should be detailed for processing details query. This paper uses simple keywords. It also uses information visualization technology for representation of the result.

Information visualization is the technology to help to understand information easily, and to support user's decision, by diagrammatize it after analyzing the information of large scale database.

3 Concept Graph-Based Retrieval

We propose the method which makes possible concept-based retrieval in extracted keywords with composition of concept graph using association and hierarchy of characteristics of hyperlink, after the keywords of web document are extracted quickly and easily by using abstraction of hyperlink information. Hyperlink is drawn up by an author of web document. It is composed of a link to referred web document and its simple explanation. The method to extract keywords using the explanation described by the author is more accurate and effective than keyword extraction through context analysis. Hyperlink between web documents also composes concept graph of keywords of each web page. Concept graph introduces information visualization to concept. It is made by connecting words only which have a particular kind of concept. It is diagrammatize with relationship of the words. In this paper, it is used for extending query or providing domain knowledge.

3.1 Concept Extracting Method

This paper uses anchor text and title tag of web document for extraction of web document keywords. Anchor text is drawn up directly by an author. All the web documents have anchor text(generality). Therefore, anchor text can be applied to keyword extraction of all the web documents. Title of web document is obtained using <Title> tag, and anchor text is obtained in the other web document linking(referring to) the web one. Anchor text is not drawn up by the author of incited document, but abstracted by an author of inciting one. Its title is extracted directly by the author of incited document. The title is only one, but anchor text is more than one in case many web documents are incited. Keywords are extracted with each words weighted. Extracted keywords can be more accurate, because they are based on not mechanical method but anchor text drawn up by author.

3.2 Query Processing

Conceptualized contents are not stored statically, but generated dynamically in execution of query. That is because the construction of concept graph for all the queries has the restriction of time and space. The method to process a query is that web documents having X as a keyword are searched for, after a particular word X is inputted. This document is defined as sibling web page. Web document which is connected in hyperlink from sibling web page is defined as hyper web page. The keyword of hyper web page is defined as hyper word. Concept graph is composed of query and the keywords with more accurate concept among hyper keywords. Query processing algorithm is as follows[Fig. 1].

```
1. input Query
2. If Query is in index
   Collect URLs with query
   Store The URLs in m_URLKey
   Else Return
3. For each URL stored in m_URLKey
   1. Obtain hyperlink information which URL has
   2. Store hyperlink information in m_DestURLKey
4. For each URL stored in m_DestURLKey
   1. Obtain keywords which URL has
   2. Store keywords in m_listKeyword(hyper keyword list)
5. List sorting by keyword frequency
6. Make string of specified format for representation of concept graph with words
   of query and concept
7. Send parameter to Java Applet
```

Fig. 1. Query Processing Algorithm

4 System Design

Basically system is composed of spider, indexer, query processor, and visual interface. Spider stores collected web pages in database, indexer extracts and indexes keywords based on contents of database, query processor processes query, and visual interface shows result of query in visual effect [Fig. 2].

A whole system is divided largely into off-line batch job and on-line processing. Off-line batch job constructs index with collected web pages, finally produces index file system. On-line processing shows result of concept graph using index file system with user's query. The processing to collect data is necessary persistently, and it takes so much time because of tremendous amount of data. The processing can be done in batch irrespective of user's response. On the contrary, system response time should be fast for query processing. Because of that, collection and query processing are separated for off-line and on-line respectively. Its whole operation is as follows: web pages are collected in off-line and stored in database, and then index file is generated through indexer. Retrieval system gets query through web server. And then query processor accesses to index file through index engine, draws up concept graph, and provides result of retrieval for user.

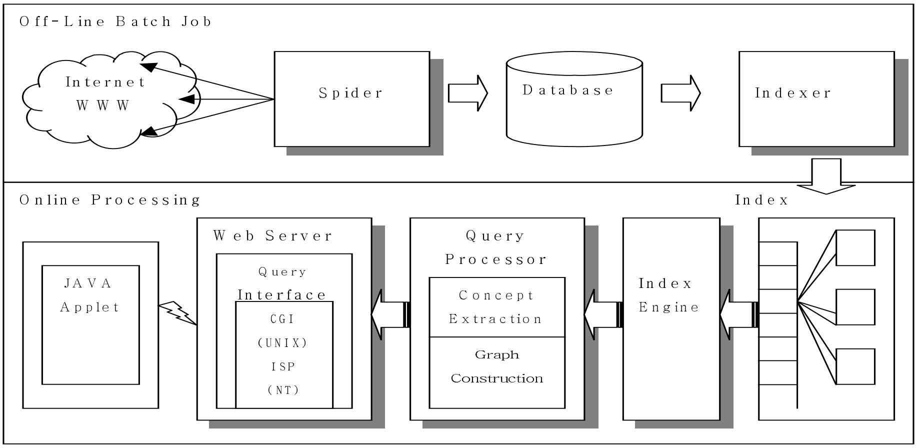


Fig. 2. System Components

4.1 Indexer

Indexer is the program which constructs index file system for retrieval service from database composed of webpage and hyperlink. If SQL statements are used for retrieval service, it takes much time for join and select, and much redundant information from database occurs for storage of hyperlink information. An unique index is constructed to reduce redundancy. Its structure is composed of indexes of URL and keyword in order to access hyperlink information quickly and easily. URL Index table plays a role of extracting quickly information for given URL, and it is made of simple hash function. Keyword index table stores information about a list of URLs with given keyword and about the number of given keyword in whole documents.

5 Experiment and Evaluation

In case of query of major concept, concept graph is composed of the words with equivalent relationship of each other, not with hierarchical relationship. That is, in case of “computer", the words of “science", "university", and "information" are represented[Figure 3]. In case of “science", that internal circle is represented large means that query connects many web pages concerned with “science". In case of “information", internal circle is represented relatively small. It means that query has not much relationship with “information", but more relationship with other word.

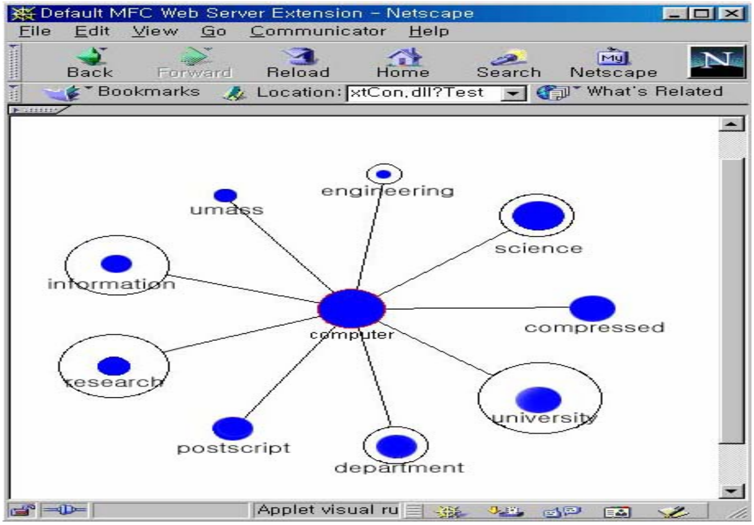

Default MFC Web Server Extension - Netscape
File Edit View Go Communicator Help
Back Forward Reload Home Search Netscape
Bookmarks Location: xtCon.dll?Test What's Related
engineering
umass
science
information
compressed
computer
research
university
postscript
department
Applet visual ru

Fig. 3. Example of Concept Graph with Major Concept Query

When it is made of the word to major concept like "computer" in keyword-based retrieval engine, a query is likely to be insignificant with so many results represented. This system, however, helps to select easily detailed concept to be searched for by providing concerned concept graph. Query of minor concept as a word concerned with detailed concept has a characteristics of low level category among concepts existing in index file.

Extracted word as keyword used frequently are not of no value unconditionally. The word of major concept like “university" appears frequently in concept graph, because it is connected to many concepts. The word of major concept, however, has no meaning for extension of query. It can be extended to query of college to study “neural network" and college for particular people. And unreachable webpages occur among URLs obtained as a result, because webpages appear and disappear so fast in web. The best advantage of this paper is to generate concept without thesaurus. It takes so much time to construct thesaurus, and method and space for storage are necessary. There is much difficulty in applying promptly all the words generated newly in the area developing rapidly like internet. With thesaurus, however, elaborate concept with no error can be extracted. It requires tremendous time and efforts to make thesaurus as in general-purpose service of large capacity like retrieval system, and it is difficult to apply the thesaurus made like that. That is because thesaurus should be made for the words of all the field. Therefore, construction of thesaurus for application of concept in retrieval system of general purpose should be automatic, or concept should be extracted in other methods. The method proposed in this paper has the restriction that document should be on web site, because concept or keyword cannot be extracted from document without hyperlink. At present, information of Korean language cannot be processed due to analysis of Korean language morpheme.

6 Conclusion

In this paper, internet retrieval method is proposed using concept graph. In this paper, hyperlink information is used to compose concept graph. Hyperlink information has useful information, because author abstracts and links concerned anchor text in hyperlink. We extract keywords of web pages using hyperlink characteristics and compose concept graph between them. Weighted multiple keywords are extracted using anchor text of abstracted information of hyperlink, and concept graph is composed by abstraction of the relationship between web pages through hyperlink into the relationship between multiple keywords. Keyword extraction method using hyperlink has an advantage to extract simple keywords by using abstraction information drawn up by an author, and experimentally keywords of web pages can be searched for relatively exactly. Keyword extraction of much more web pages is possible, because it can be extracted by hyperlink information of web pages only without processing main text of particular web pages for keyword extraction. Concept graph has an advantage to represent domain knowledge of a real world without thesaurus by representing relationship between keyword nodes. User can extend query without enough knowledge about it. Its disadvantage is that unnecessary information is represented in concept graph with redunancy of keywords without stemming. Concept cannot be also extracted from documents without hyperlink information. For future work, more effective method which is not simple frequency, and multiple query processing, need to be researched in selection of concerned words for concept graph. The method is also necessary for drawing multiple-stepped graph as well as single-stepped graph.

References

1. Michael J. A. Berry, Gordon Linoff, "link analysis", Data Mining Techniques: For marketing, Sales, and Customer Support, Wiley Computer Publishing, (1998) 216-242
2. Sergey Brin, Lawrence Page, "The Anatomy of a Large-Scale Hyper textual Web Search Engine", In Proceeding of the seventh International World Wide Web Conference,(2002)
3. James P. Callan, W. Bruce Croft, Stephen M. Harding, "The INQUERY Retrieval System", Database and Expert Systems Applications, (1992) 78-83

4. Jeromy Carrie, Rick Kazman, "WebQuery: Searching and Visualizing the Web through Connectivity", In Proceeding of the Sixth International World Wide Web Conference, (1997)
5. Hsinchun Chen, Chris Schuffels, Rich Orwig, " Internet Categorization and search: A Self Organizing Approach", Journal of Visual Communication and Image Representation, Vol. 7, (1996) 88-102
6. Jon M. Klienberg, "Authoritative Sources in a Hyperlinked Environment", In Proceedings of the ACM-SIAM Symposium on Discrete Algorithms, (1998) 668-677
7. M. Koster, "ALIWEB: Archie-like indexing in the web", Computer Networks and ISDN Systems Vol. 27, (1994) 175-182
8. Massimo Marchiori, " The Quest for Correct Information on the Web: Hyper Search Engines", In Proceeding of the Sixth International World Wide Web Conference, (1997)
9. Mauldin, Leavitt, "Web-agent related research at the CMT", In Proceedings of the ACM Special nterest Group on Networked Information Discovery and retrival, (1994)
10. Christian Neusess, Robert E. Kent, "Conceptual Analysis of Resource Meta-information", Computer Networks and ISDN Systems, Vol. 27, (1995) 973-984
11. G. Salton, "Developments in automatic text retrieval", Science, Vol. 253, (1991) 974-979
12. Ron Weiss, Bienvenido Velesz, Ma□ A. Sheldon, "HyPursuit: A Hierarchical Network Search Engine that Exploits Content-Link Hypertext Clustering", ACM Conference on Hypertext, (2001) 180-193
13. Budi Yuwono, Kit L. Lee, "Search and Ranking Algorithms for Locating Resources on the World Wide Web", International Conference on Data Engineering, (2000) 164-171

Development of Integrated DAO Pattern Applying Iterator Pattern

Seong-Man Choi, Cheol-Jung Yoo, and Ok-Bae Chang

Dept. of Computer Science & Statistical Information, Chonbuk National University, 664-14,
1Ga, Duckjin-Dong, Jeonju, Jeonbuk, 561-756, South Korea
{sm3099, cjyoo, okjang}@chonbuk.ac.kr

Abstract. EJB, providing specification for development and deployment of component based application, permits distributed development as a central element of J2EE environment that automatically manages transaction management, persistence, and concurrency control, which are the most complicated components in an enterprise environment. In this paper, we aim to resolve DAOs transaction logic complexity and performance reduction of components in the EJB based legacy system. Therefore, this paper describes the design and implementation of IDAO that applies iterator pattern. IDAO achieves an effect that reduces the complexity of transaction logic, system overload by database connection, and reduction of performance through container managed transactions.

1 Introduction

A component system which is highlighted at the present time divides a large and complex problem into small parts by a certain standard and solves them first then constructs a solution from the simple bases. Such a component development transacts the affairs and their related data as one unit according to the object-oriented theory. A need for this originated from the constraints of present technology which can't adapt to the changes of business environment and technology, and the problems of business integration or distributed environment, portability, need of light weight, ability of a developer and non-automated development process. The main purpose of the component development is reuse of an existing design for other situations repeatedly to facilitate a remarkable improvement in development productivity and quality. It also increases its reliability with its easiness of exchange when the center control is difficult and by improving productivity by reuse of a design and its implement and using an efficient tested code. Java newly produced EJB(Enterprise JavaBeans), a next generation standard of component technology for enterprise application. In this paper, DAO(Data Access Object) which encapsulates the access of the EJB-based legacy system will be studied. Iterator pattern applied IDAO(Integrated DAO) also will be recommended to solve complexity, a problem caused by the legacy system, of DAO transaction logic, unnecessary formation of DAO and system overloading and to access the database efficiently. The IDAO is implemented for an efficient access to the database which can reduce the complexity of transaction operation through container management transaction. So, it reduced overloading and degradation of the system performance.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 955 – 965, 2006.

This paper has been organized as follows: Section 2 focuses on DAO in the legacy system and problems from DAO. Section 3 focuses on the application process of iterator pattern and design of IDAO. Section 4, a workflow of IDAO implementation process shows the implementation process of iteratpr pattern applied IDAO and the implementation of IDAO where the iterator pattern is actually applied will be explained. The last section 5 provides conclusions and outlines future works.

2 Limitations of the DAO Pattern

A purchasing process of a product after a buyer connects to the product server of a shopping mall through Web-browser at e-commerce. A customer connects to the product server and reads information on products, purchase form, offered by the shopping mall. Among the purchasing process, each necessary DAO in the product purchasing process after a customer connects the products server of the shopping mall through Web browser will be studied. First, a user puts his/her ID and password in for conformation. Then Account DAO checks if they are matched with the ones in Account DB. If they are matched then displays a product catalog on the screen. If not, a checking process for confirmation is conducted again. In catalog of products list, Catalog DAO of products catalog access Catalog DB and retrieves the corresponding data. To check the stock of the wanted products, Inventory DAO in the repository accesses to Inventory DB, checks if the products are in the stock and transfers the information to the user then he/she can order them. At the time, Order DAO accesses Order DB and preserves the information on the purchasing in it. Figure 1 shows a process through which the information preserved in Cart DB is brought and displayed using stateless session bean because at the time the information is inquired only.

Therefore, each corresponding DAO will be studied. DAO expresses XML and an object for an individual purpose of database vender. DAO is used to encapsulate access to the database and is suitable for a large scale arrangement. It implements an individual vendor and resource and makes it move easily from the bean management persistence to the container management persistence. An example of the corresponding contents to

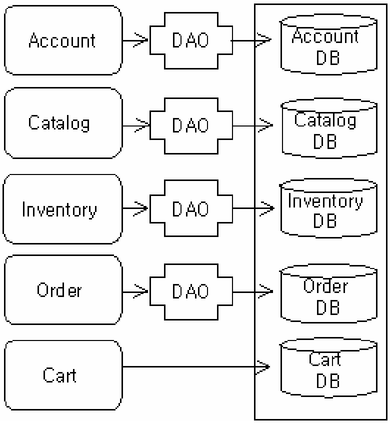


Fig. 1. DAO structure of legacy system

DAO architecture of legacy system, as noted in Figure 1, Catalog DAO source will be studied. Catalog DAO class Figure 2 encapsulates all the SQL produced by Catalog EJB and maps the data preserved in database as a necessary object for Catalog EJB.

```
public class CatalogDAO {
   private Connection con;
   public CatalogDAO(Connection con) {
      this.con = con;
   }
   public Category getCategory(String categoryId)
   throws SQLException {
    String qstr = "select catid, name, descn
    from " + DatabaseNames.CATEGORY_TABLE
    + " where catid = '" +
     categoryId + "'";
      Statement stmt = con.createStatement();
      ResultSet rs = stmt.executeQuery(qstr);
      Category cat = null;
      while (rs.next()) {
         int i = 1;
         String catid = rs.getString(i++).trim();
         String name = rs.getString(i++);
         String descn = rs.getString(i++);
         cat = new Category(catid, name, descn);
      }
      rs.close();
      stmt.close();
      return cat;
   }
     ........................

public class CatalogImpl implements CatalogModel {
   public Category getCategory(String categoryId) {
      Connection con = getDBConnection();
      try {
         CatalogDAO dao = new CatalogDAO(con);
         return dao.getCategory(categoryId);
      } catch (SQLException se) {
         throw new GeneralFailureException(se);
      } finally {
         try {
            con.close();
         } catch (Exception ex) {
        throw new GeneralFailureException(ex);
  ......................
```

Fig. 2. Catalog DAO architecture

Catalog entity bean is offered by Catalog EJB which is inherited from CatalogImpl. It is the first screen list, presented to a user after he/she enters the user ID and password and allows him/her to choose products. To display a product list for the user, the database is accessed and each necessary method is prompted for information on the products through Catalog DAO. First, information on categoryid, name, descn through getCategory(), information on product list, price list and prices through getProduct() and information on ID of the chosen products, price list and price through getItem() is acquired. The information on the chosen products is repeatedly conducted through searchProducts() and notifies the user of the present condition of the products, in particular whether they are in stock or out of stock. getQueryStr() shows information on selling, returning, restoration and all of the goods which is information of all selling division, input at icons for the user's convenience.

Construct of the whole components can be predicted by mentioning corresponding contents of catalog entity bean, a part of the purchasing process of the user in a e-commerce site, drawn in Web application. The whole set of components consists of several components and each component accesses the database through its DAO. This causes some problems. First, when DAO operates the corresponding data the operation is conducted at different transactions, which induces complexity of the transaction logic and causes system overloading. Second, DAO connected to each bean forms unnecessary DAO as it becomes instant when the bean becomes instant. Third, it decreases performance of the system by occupying a lot of memory with unnecessary DAOs.

3 Design of IDAO Applying Iterator Pattern

3.1 Process of Applying Iterator Pattern

Design pattern is a record of information on a design defined well in the past, which can be applied again for a similar situation in the future [5]. If a design pattern is applied to a design of EJB component, designing time and communication time between developers are reduced and safety and performance of the system is increased by applying an efficient and useful design pattern. Gamma divides the design pattern into production, pattern, construct and behavior pattern. If it is divided for a purpose which reflects what the pattern does and according to a defined range when the pattern is applied first to an object or a class.

The IDAO offered in this paper is an applied iterator pattern, a behavior pattern. It is a most useful pattern and one of the simplest patterns. It claims a method and defines interface to access an object in the order in a collection. A class which accesses the collection through the interface exists individually while implementing class of the interface. As a result, if the iterator pattern is applied, it can offer a method defined through a collection of data elements without displaying the conducting method of the collection [8]. The IDAO offered in this paper is an applied iterator pattern. Figure 3 shows the process of being component of the iterator pattern [9].

The component is checked by requirements and constraints of a special application and design pattern chooses a suitable pattern for interaction with the component. The instance formation information preparing stage of design pattern includes relation

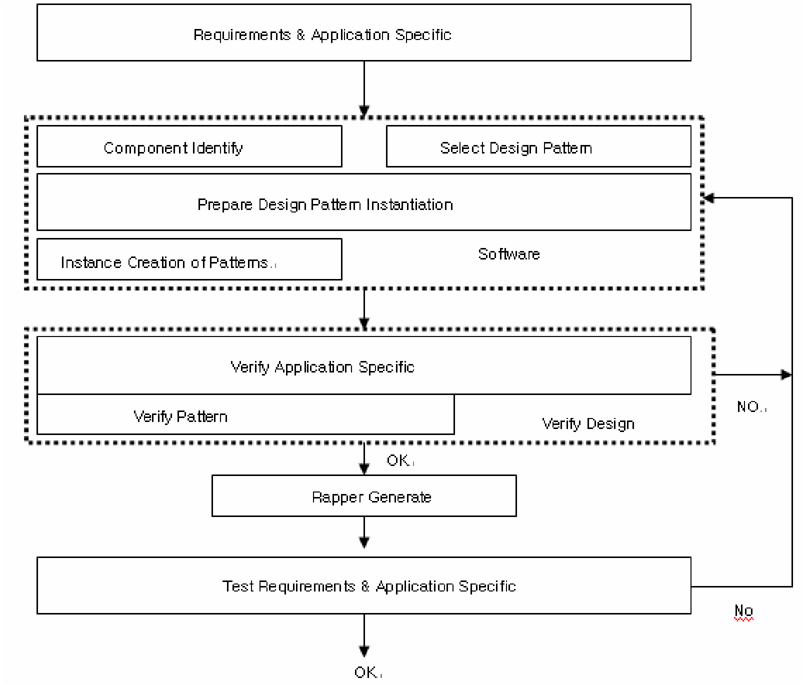

Requirements & Application Specific
Component Identify
Select Design Pattern
Prepare Design Pattern Instantiation
Instance Creation of Patterns.
Software
Verify Application Specific
Verify Pattern
Verify Design
NO.
OK.
Rapper Generate
Test Requirements & Application Specific
No
OK.

Fig. 3. Component process of iterator pattern

between design pattern elements, design elements and chosen design pattern. Design pattern elements include the events caused by the component and services offered by the component and these are related to the elements included in the design pattern. Instance formation of the pattern is used to change the design pattern to abstract solution in the part of software design specification [10]. The design verification stage is divided into special application constraints verification and pattern coherence verification. First, at the special application constraints verification stage, coherence of the structure, interaction are verified and service offered by the component is also verified. Next, at the stage of pattern coherence verification, whether or not the design pattern instance formation has the same constraints as the first design pattern is verified. At the wrapper formation stage, it acts as a component decorator. Interaction between all the components is conducted through the wrapper. Component process of iterator pattern Figure 3 is adapted to IDAO offered in this paper. Figure 4 presents the results of this study.

User interface classes consisted of IDAOs are explained as combination class of IDAO Figure 4. The instance of IDAO requires expression of IDAOItem objects in the collection encapsulated by IDAOCollection objects. Objects of IDAO can't access objects of IDAOCollection. Instead, IDAO gets an object implementing IDAO IteratorIF interface which defines a method of bringing contents of IDAOItem objects caused as a result. Figure 5 shows an applied source of iterator pattern applied to IDAO based on figure 4. Skeleton code is shown for implementation of the design.

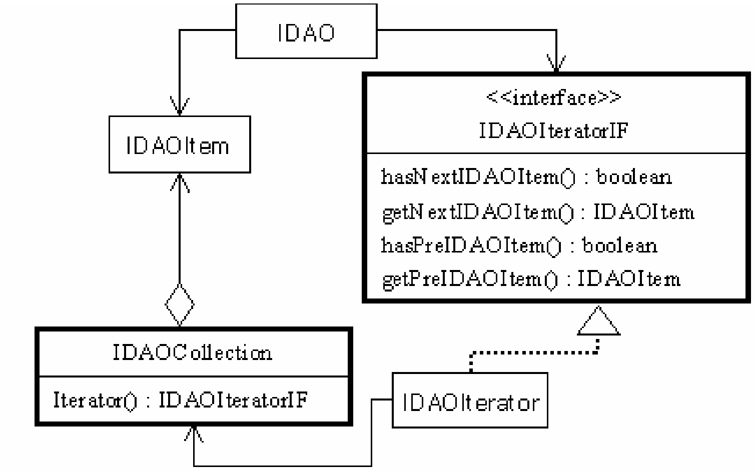

IDAO
IDAOItem
<<interface>>
IDAOIteratorIF
hasNextIDAOItem() : boolean
getNextIDAOItem() : IDAOItem
hasPreIDAOItem() : boolean
getPreIDAOItem() : IDAOItem
IDAOCollection
Iterator() : IDAOIteratorIF
IDAOIterator

Fig. 4. Structure of iterator pattern applied IDAO

```
public interface IDAOIteratorIF {
  public boolean hasNextIDAOItem();
  public IDAOItem getNextIDAOItem();
  public boolean hasPreIDAOItem();
  public IDAOItem getPreIDAOItem();
  }
public class IDAOCollection {
  public IDAOIteratorIF iterator() {
    return new IDAOIterator();
  } // iterator()
  public class IDAOIterator implements IDAOIteratorIF {
    public boolean hasNextIDAOItem() {
    ......
    } // hasNextIDAOItem()
    public IDAOItem getNextIDAOItem() {
    ......
    } //getNextIDAOItem()
    public boolean hasPreIDAOItem() {
    ......
    } // hasPreIDAOItem()
    public IDAOItem getPreIDAOItem() {
    ......
    } // getPreIDAOItem()
  } // class IDAOIterator
  ......
} // class IDAOCollection
```

Fig. 5. Iterator pattern applied to IDAO

When the skeleton list of IDAOCollection class is examined, it includes iterator() which is used to get repeated objects on the contents of IDAOCollection object. IDAOCollection class also includes instance private class of iterator method. It is possible to make one DB connection and reduce necessarily formed instance of an object according to the kinds of data by offering IDAOIteratorIF, single interface which can access various kinds of data using iterator as above. In other words, in the previous case, access was conducted using many objects, such as Account DAO, Catalog DAO, Inventory DAO and Order DAO, according to the kinds of data even though the access to the data was the same but now it is simplified as IDAO. The amount of transaction processing can be reduced by allocating the transaction which causes transaction overloading in IDAO methods only according to each DAO method by using an iterator pattern applied IDAO.

3.2 Design of IDAO Applying Iterator Pattern

Iterator pattern applied IDAO is through component process of design pattern. It is a structure of improved IDAO to solve complex transaction logic, unnecessary DAO formation and system overloading. When the IDAO pattern is accessed at each component, the IDAO which has a character of each DB accesses corresponding DB in real-time. Use of combined IDAO reduces complexity of transaction operation in container management transaction more than operating contents of numbers of DAOs in transaction, so as a result it can reduce system overloading. It is also expected to increase performance, scalability and efficiency.

4 Implementation of IDAO Applying Iterator Pattern

4.1 Implementation Process Workflow of IDAO

Implementation process workflow which was applied to implement EJB-based IDAO present in this paper is shown Figure 6.

Boxes are a series of action, workflow, producing meaningful artifacts. First, if a user's requirement is transferred to requirements definition workflow, it gets usecase model and business conceptual model as a result of a previous stage and transfers them to specification workflow. The specification workflow, then, forms component specification and component architecture using information of technological constraints such as legacy system, package, database, special architecture or use of the tools. The results of specification workflow are used in component supply workflow to determine component purchase or development. They also offer guide lines for exact combination of the components in the assembling workflow and are used as necessary inputs for formation of test scenario in the test workflow. In the supply workflow, a component is developed, purchased from the third-party component supplier or a necessary component can be used by reuse, integration and refining of a legacy component or a software. Unit test on a component is conducted before the component assembling in the workflow. A component is properly integrated with the legacy system or user interfaces to make a suitable application for a user's requirements in the

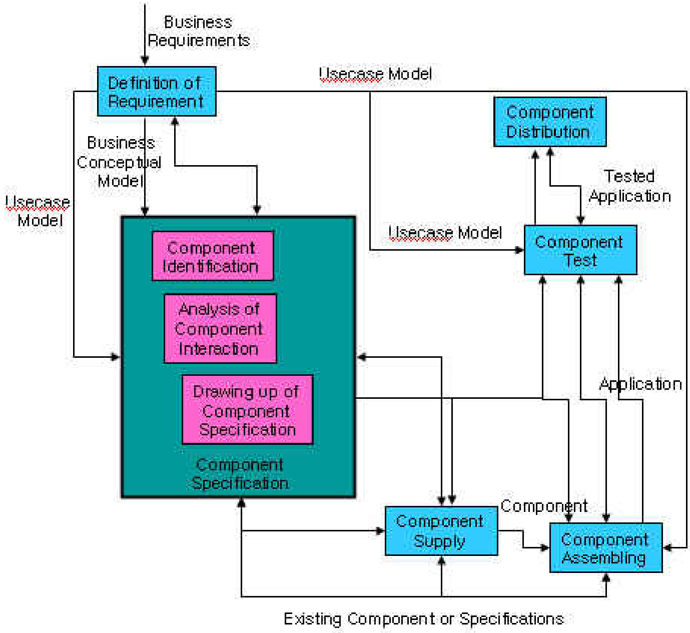


Fig. 6. Implementation process workflow of IDAO

component assembly workflow. Various types of testing, such as unit testing, module testing, component testing which are components, groups of components, server system or whole system, are conducted in the component test workflow before appropriate data for users' requirements are transferred. After preparation is conducted for distribution of the component, when the information of server location with a help of distribution wizard all the works related to distribution such as drawing up of developer descriptor are automatically conducted.

4.2 Implementation of IDAO Applying Iterator Pattern

A part of the source of iterator pattern applied IDAO is shown in Figure 7. The source improves performance of entity bean by integrating the access code of a component implemented in J2EE platform by applying an iterator pattern to the data.

In other words, frequently formed connection of DB is maximum suppressed during a life cycle of the entity bean by including the access code. In the case of transaction, database connection of DAO is united to getConnection() method to reduce processing units of transaction. The management of the connection is delegated to the Web application server and it causes pooling of the connection. As a result, this reduces performance degradation of the system and overloading by reducing database connection number. The class applied in IDAO also makes input objects as a filter form using methods such as setInput() and getInput(). It requires the work that is to be

```
package IDAOClass

public class DAOClass {
  public DAOClass() {
    }
  public IDAO getInput(Object obj) {
    }
  public IDAO getOutput(Object obj) {
    }
  public void setInput(Object obj) {
    }
  public void setOutput(Object obj) {
    }
  }
```

```
package IDAOClass;

import java.rmi.RemoteException;
import javax.rmi.PortableRemoteObject;
import javax.naming.InnitialContext;
import javax.ejb.CreateException;

public class IDAOFactory extends IDAOIterator {
  IDAO iDao = null;
  DAOClass daoclass = null;

  public static Connection getConnetion() throws        javax.naming.NamingException {
    InitialContext initial = new InitialContext();
    Object objref =                          initial.lookup(JNDINames.CONNECTION_EJBHOME);
    return (Connection)
      PortableRemoteObject.narrow(objref,
        Connection.class);
  }

public static IDAO getDAOClass(DAOClass daoclass) {
  iDAO = daoclass.getInput(daoclass);
  return (IDAO) iDAO;
  }
public static void create(IDAO iDAO) {
  iDAO.create(iDAO.GetInput(daoclass));
  }
public static void store(IDAO iDAO) {
  iDAO.store(DAO.GetInput(daoclass));
  }
public static void remove(IDAO iDAO) {
  iDAO.remote(iDAO.GetInput(daoclass));
  }
}
```

Fig. 7. Iterator pattern applied IDAO source

conducted through many DAOs to be processed through only one method and doesn't cause unnecessary DAO formation. Thus, it improves extension and efficiency of the system.

4.3 Assessment of IDAO

The IDAO was offered to solve complexity of DAO which is used to encapsulate DB access in the legacy system, unnecessary DAO formation and system overloading. The IDAO is applied iterator pattern which announces a method to access objects in order in collection. As a result of applying the iterator pattern, a class accessing simply the collection through an interface exists individually with a class which implements the interface. So, the IDAO offers a defined method through a collection of data elements without showing a conducted method on the data collection. The IDAO is compared with legacy system on performance measure elements, and the results of this comparison are presented in Table 1. The degree of system overloading, complexity of transaction and efficiency of the method depend on the number of DB connection, transactions and DAOs.

Table 1. Legacy system and performance measure elements

Pattern Contents	DAO	IDAO
number of DB connection	number of DAO object	one
number of transaction processing	number of DAO object method	IDAO method
number of DAO	number of DAO object	one

Table 2. Performance comparison of DAO and IDAO

Pattern Contents	DAO	IDAO
system overloading	high	relatively low
complexity of transaction logic	high	low
call time	long	short
memory efficiency	bad	good
unnecessary DAO productivity	yes	no
performance improvement and scalability	no support	support

When the DAO of legacy system is compared with IDAO based on Table 1, a performance comparison table such as Table 2 can be made. The number of methods which participates in transaction is reduced. Unnecessarily repeated logic is reduced by not having many DAO objects. The IDAO improves system performance by minimizing complexity of transaction operation, unnecessary DAO formation and system overloading by having one DB connection.

5 Conclusions and Future Works

The Iterator pattern, behavior pattern of design pattern, applied IDAO is discussed in this paper to reduce the complexity of transaction logic, unnecessary DAO formation and system overloading that occur in DAO which encapsulates access of DB in EJB-based legacy system. The IDAO reduces the complexity of transaction through a container management transaction for efficient access of DB and reduces system overloading and degradation. The design of IDAO which can be applied in an e-business system with new Java technology such as JMS or connector architecture and a design of optimized IDAO related with a special product such as Web-Logic of BEA must still be conducted.

References

1. Clemens Szyperski, Component Software : Beyond Object-Oriented Programming, Addison Wesley Lognman, Inc.(1998)
2. Peter Herzum and Oliver Sims, The Business Component Approach : Business Object Design and Implementation Ⅱ, OOPSLA ‘98 Workshop Proceedings, UK : Spring-Verlag(1998)
3. Desmond F. D Souza and Alan Cameron Wills, Objects, Components and Frameworks with UML : The Catalysis Approach, Addison Wesley Longman, Inc.(1999)
4. Deepak Alur, John Crupi and Dan Malks, Core J2EE Patterns : Best Practices and Design Strategies, Prentice Hall PTR(2001)
5. Mark Grand, Patterns in Java, Volume 1 : A Catalog of Reusable Design Patterns Illustrated with UML, John Wiley & Sons, Inc.(1998)
6. Karl Rege, Design Patterns for Component-Oriented Software Development, in Proceedings of EUROMICRO(1999) 220-228
7. Erich Gamma, Richard Helm, Raplh Johnson and John Vlis-sides, Design Patterns : Elements of Reusable Object- Oriented Software, Addison Wesley Longman, Inc.(1995)
8. Cooper and James William, Java Design Patterns : A Tutorial, Addison Wesley & Sons, Inc.(2000)
9. Stephen S. Yau and Ning Dong, Integration in Component- Based Software Development Using Design Patterns, in Proceedings of COMPSAC(2000) 369-374
10. John Cheesman and John Daniels, UML Components - A Simple Process for Specifying Component-Based Software, Addison Wesley Longman, Inc.(2001)

A Coupling Metric Applying the Characteristics of Components

Misook Choi[1] and Seojeong Lee[2]

[1] Woosuk University, 490, Hujong-ri, Samnye-up,
Wanju-kun, Chonbuk, Korea
khc67_kr@hanmail.net
[2] Korea Maritime University, College of Engineering,
Devision of Information and Technology,
Department of Computer Engineering, Busan, Korea
sjlee@bada.hhu.ac.kr

Abstract. A high coupling between components makes it difficult to build the modulation of software and the reuse of components, and to manage the system due to the ripple effect by software change impact. Thus, a coupling metric is required to measure the coupling between components in order to design software effectively. In this paper, we propose an approach to improving the existing component-based coupling metrics by considering the dependency about the structured relationships and the method call types between classes. In addition, we prove the theoretical soundness of the proposed metric by the axioms of briand et al. and suggest the accuracy and practicality of the proposed metric through a comparison with the conventional metrics.

1 Introduction

The high coupling between components makes it difficult to build the modulation of software and the reuse of components, and to manage the system due to the ripple effect by software change impact. That is, the more coupling between components is low, the more software in maintenance phase can be managed effectively and components can be reused effectively. Moreover, a quantitative measuring method that uses a software metric, can assist developers in finding a potential defect in the software development phase, and to correct the defects[2]. Thus, a coupling metric is required to measure the coupling between components in order to design a more independent component effectively. Existing coupling metrics to measure the quality of components are not measured precisely, because they are applied without any modification or with some modification of object-oriented metrics. Accordingly, a coupling metric applying the characteristics of components is required.

In this paper, we propose an approach to improving the existing component-based coupling metrics by considering the degree of static dependency by the structured relationships between classes and the degree of dynamic dependency by the method call types between classes. The proposed coupling metric measures the coupling of

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 966 – 975, 2006.

components precisely. Our coupling metric is used to identify more independent components and to assist designing of high quality components in software development phase. It reduces software development time and efforts.

2 Related Works

The CBO (Coupling Between Object Classes) and RFC (Response For a Class) by Chidamber-Kemerer[3] are widely known, and are object-oriented coupling metrics. The CBO can be defined as the number of other classes that are related to a class. When the RS is a Response Set, RFC can be defined as RFC = |RS|, in here, $RS = \{M\} \bigcup all_i \{R_i\}$. $\{M\}$ is a set of method, which is defined in the class, $\{R_i\}$ is a set of method, which is called by the method of i. The more number of called methods are increased, the more performing a test and debugging of a class is complicated, and maintaining is also difficult. A representative object-oriented coupling metric proposed by Henderson-Sellers[4] is the MPC (Message Passing Coupling). The MPC is defined as a number of send operations in a class. A greater increase in sending operations defined in a class leads to a more complex test and debugging process, and also to difficult maintenance.

A component-based coupling metric proposed by Lee[5] is used to identify components, and it is defined by a static coupling metric applying static relationships between classes and a dynamic coupling metric applying the number of method calls between classes. Koh[6] defines a component-based coupling metric based on the fact that how the classes between the components is shared by the methods to perform functions. Therefore, the existing component-based coupling metrics are very similar to object-oriented coupling metrics.

But there are wide structural differences between the object-oriented system with classes being a unit of reusability, and the component-based system in which a component with a group of classes closely related to each other becomes a unit of reusability. Also, there are wide differences in the unit of function. That is, a class executes functions by referring the included methods and attributes in the class, but a component executes functions by interactions between the included classes in the component. In addition, the function of components is to be implemented by referring to the creation, deletion, and modification of internal or external classes because the component consisted of a number of classes. Therefore, the component–based coupling metric must be regarded, not only as the number of classes that are related between components, but also the type and number of message calls. That is, measuring the component's coupling with object-oriented metrics themselves is inadequate. Currently, some component coupling metrics[5,6] exist. But most of them are applied without any modification or with some modification of object-oriented metrics[3,4]. Accordingly, component metrics require the application of the characteristics of components for measuring the coupling accurately. Therefore, we propose a coupling metric applying the static dependency by a structure of classes and the dynamic dependency by the number and types of method call between classes.

3 A Component-Based Coupling Metric

This chapter defines the static and dynamic dependency between classes and a component-based coupling metric applying them, and suggests the theoretical validity of the proposed coupling metric according to a framework proposed by Briand [7].

3.1 A Static and Dynamic Dependency Between Classes

This section defines types of the static and dynamic dependency between classes, to measure the component coupling more precisely. They are applied to our component-based coupling metric.

[Type 1] ***The strength of static dependency according to structural relationships between classes follows the rank of aggregation > inheritance > association.***

[Type 2] ***If classes are in an association relationship, the strength of dynamic dependency according to dynamic relationships between classes is different according to method call types of create, delete, write, and read.***

Method call types can be classified as follows; ***<A, B, R>***, where ***A*** and ***B*** are classes, and ***R*** is relationship.

1. In case of a class ***A*** sends a message to create the object of a class ***B***.
2. In case of a class ***A*** sends a message to delete the object of a class ***B***.
3. In case of a class ***A*** sends a message to write in object of a class ***B***.
4. In case of a class ***A*** sends a message to read the object of a class ***B***.

In the case of 1 and 2, the structure of the object in class ***B*** is changed by class ***A***, so that other objects referring to the object of class ***B*** are totally influenced in structural aspect. In the case of 3, the other objects referring to the object of class ***B*** are impacted only in values because the values of the object in class ***B*** are altered only without changing its structure. In the case of 4, the other objects referring to the object in class ***B*** do not undergo any influences because the structure and state of the object in class ***B*** by class ***A*** is not changed.

[Type 3] ***Even if bi-direction method call exists between classes, the strength of dependency between classes is different according to method call types.***

If ***<A, B>*** $\in$ ***R*** and the object of class ***A*** sends a message to create the object of class ***B,*** then ***B*** is dependent on ***A***. The creation of the object of class ***B*** structurally affects other objects manipulating the class ***B***. However, if the class ***A*** sends a message to read the object of class ***B***, the class ***A*** does not affect the object of class ***B*** at all because the class ***A*** is simply reading the data of the object by class ***B***. The dependency relationships between classes rely on a direction of method calls by the method call types. Therefore, we must consider method call types by the direction of method calls. The dependency types by the direction of method calls are as follows.

1. In case that the class ***A*** invokes methods on the class ***B*** or the class ***B*** invokes methods on the class ***A.***
2. In case that the class ***A*** and ***B*** invoke methods on each other.

In 1, the strength of dependency between classes relies on types of method calls. In 2, although the existing research[3] defines that there is unconditionally strong dependency in the presence of bi-directional method calls, this paper defines that the strength of dependency depends on the method call types by type 2.

[Type 4] ***The connectivity strength between classes according to the number of method calls increases linearly.***

The connectivity strength between classes is different according to the number of method calls. Namely, between classes, several times of method calls have much higher connectivity strength than only one time of method call has. Therefore the connectivity strength between classes by the number of method calls increase linearly.

3.2 A Component-based Coupling Metric

This section defines a component coupling metric using the static and dynamic dependency types between classes defined in the section 3.1.

[Definition 1] ***Components in a System***

A system consists of finite components. If the system is referred to as S and the involved components in the system S are referred to as $BC_i (i = 1 \ldots l)$, the system S is defined as follows.

$$S = \{BC_1, BC_2, \ldots, BC_l\}$$

[Definition 2] ***Classes in a Component***

Since the components of a system, $BC_i (i = 1...l)$ are composed of the limited number of classes C , *components* $BC_i (i = 1...l)$ are defined as follows.

$$BC_i = \{C_{i1}, C_{i2}, ..., C_{im}\}$$

[Definition 3] ***Methods of the Class***

A component consists of a group of classes and interactions between classes or components depend on the method calls in which each class includes. So, *the methods of each class* $M(C_j)(j = 1..k)$ are defined as follows.

$$M(C_j) = \{m_{j1}, m_{j2}, ..., m_{jn}\}$$

[Definition 4] ***Method Calls between Classes***

Interactions between classes depend on method calls between classes. If methods $m' \in M(C_g), m \in M(C_y)$ exist for different classes $C_g (1 \le g \le m), C_y (1 \le y \le m)$, and *the method* m' *calls* m *or the method* m' *is called by the method* m, it is defined as (m', m) .

[Definition 5] ***Interactions by Types of Method Call between Classes***

Interactions between classes depend on types of method call between classes.

If methods $m' \in M(C_g), m \in M(C_y)$ exist for different classes $C_g (1 \le g \le m)$, $C_y (1 \le y \le m)$, interactions by the types of method call between classes are defined as follows.

$C(m', m)$: In case that the class C_g and class C_y ***send a message to create*** *data each other.*

$D(m', m)$: In case that the class C_g and class C_y ***send a message to delete*** *data each other.*

$W(m',m)$: In case that the class C_g and class C_y ***send a message to write*** *data each other.*

$R(m',m)$: In case that the class C_g and class C_y ***send a message to read*** *data each other.*

[Definition 6] ***A Coupling between Classes(CC)***

The structural relationship between classes includes association, generalization, and aggregation relationship, and the connectivity strength of static dependency by the structural relationship between classes is different according to types of each structural relationship by Type 1of 3.1. But the connectivity strength of static dependency by the structural relationship between classes is decided to the types of method call. For example, in a aggregation relationship (that is, class A contains classes, B and C), if an instance of class A is created, deleted, written, or read, the same operation is applied to the instances of classes, B and C. That is, in a aggregation relationship, invoking an operation on an instance of class A has a direct impact on the instances of classes, B and C. The inheritance relationship is the same. Accordingly, the degree of static and dynamic dependency between classes is decided to the number of method calls and types of method call: create, delete > write > read.

Thus, the coupling between classes can be defined by applying the weights according to the types of method call(create, delete > write > read), and by considering the number of method calls. If methods $m_i \in M(C_i), m_j \in M(C_j)$ exist for different classes $C_i(1 \le i \le m), C_j(1 \le j \le m)$, a coupling metric CC between classes is as follows.

$$CC(C_i, C_j) = WM_c \sum_{m_i \in M(C_i), m_j \in M(C_j)} C(m_i, m_j) + WM_d \sum_{m_i \in M(C_i), m_j \in M(C_j)} D(m_i, m_j)$$
$$+ WM_w \sum_{m_i \in M(C_i), m_j \in M(C_j)} W(m_i, m_j) + WM_r \sum_{m_i \in M(C_i), m_j \in M(C_j)} R(m_i, m_j)$$

Where :

$WM_c\,(create), WM_d\,(delete), WM_w\,(write), WM_r\,(read)$: *Weights about types of method call*

Therefore, the coupling metric between classes means the sum of the multiplications of the number of method calls according to the types of method call between classes and the weights according to the type of method call, respectively.

[Definition 8] A ***CouPling of a Component (CPC)***

The coupling of a component BC_k , $CPC(BC_k)$ is defined by applying ***a Coupling between Classes(CC)***, a coupling of a component is as follows.

$$CPC(BC_k) = \sum_{C_i \in BC_k, C_j \notin BC_k} CC(C_i, C_j)$$

[Definition 9] ***A Coupling Between Components (CBC)***

The coupling of $CBC(BC_l, BC_m)$ between components for the two components of BC_l *and* BC_m is defined by the couplings for the classes of $C_l \in BC_l$ *and* $C_m \in BC_m$, which are included in each component. Therefore, the coupling between components is defined as follows.

$$CBC(BC_l, BC_m) = \sum_{C_l \in BC_l, C_m \in BC_m} CC(C_l, C_m)$$

3.3 A Theoretical Soundness of the Proposed Coupling Metric

This section suggests the theoretical soundness by applying a framework proposed by Briand et al to our coupling metric.

[Property 1] ***Non-Negativity***

$$\boxed{(\forall BC_k)\ ,\ CPC(BC_k) \geq 0}$$

Proof: $CPC(BC_k) = \sum_{C_i \in BC_k, C_j \notin BC_k} CC(C_i, C_j) \geq 0$

[Property 2] ***Null value***

$$\boxed{(for\ \forall BC_k\ \ such that\ \exists\ \forall C_{ki} \in BC_k\ \ and\ \forall C_j \notin BC_k)\ , CC(C_{ki}, C_j) = 0\ \ \rightarrow CPC(BC_k) = 0}$$

Proof: If $CC(C_{ki}, C_j) = 0$, then $CPC(BC_k) = 0.$

[Property 3] ***Monotononicity***

$$\boxed{\begin{array}{l}(for\ \forall BC_k\ \ such\ that\ \exists\ \forall C_{ki} \in BC_k\ \ and\ \forall C_j \notin BC_k)\ , \\ CC(C_{ki}, C_j) \leq CC'(C_{ki}, C_j)\ \ \rightarrow CPC(BC_k) \leq CPC'(BC_k)\end{array}}$$

Proof: If $CC(C_{ki}, C_j) \leq CC'(C_{ki}, C_j)$,

then $\sum_{C_{ki} \in BC_k, C_j \notin BC_k} CC(C_{ki}, C_j) \leq \sum_{C_{ki} \in BC_k, C_j \notin BC_k} CC'(C_{ki}, C_j)$.

Hence $CPC(BC_k) \leq CPC'(BC_k)$.

[Property 4] Merging of components

$$\boxed{\begin{array}{l}(for \forall BC_i, \forall BC_j, \forall BC_k\ \ such\ that\ \exists\ \forall C_i \in BC_j\ \ and\ \forall C_j \in BC_k)\ , \\ BC_i = BC_j \cup BC_k\ \ and\ \ CC(C_i, C_j) \neq 0\ \ \rightarrow \sum\{CPC(BC_j), CPC(BC_k)\} \geq CPC(BC_i)\end{array}}$$

Proof:

$$\begin{aligned} CPC(BC_i) &= \sum_{C_i \in BC_i, C_j \notin BC_i} CC(C_i, C_j) = \sum_{C_i \in BC_j \cup BC_k,\ C_j \notin BC_j \cup BC_k} CC(C_i, C_j) \\ &= \sum_{C_i \in BC_j, C_j \notin BC_i \cup BC_k} CC(C_i, C_j) + \sum_{C_i \in BC_k,\ C_j \notin BC_i \cup BC_k} CC(C_i, C_j) \\ &\leq \sum_{C_i \in BC_j, C_j \notin BC_j} CC(C_i, C_j) + \sum_{C_i \in BC_k,\ C_j \notin BC_k} CC(C_i, C_j) \\ &= CPC(BC_j) + CPC(BC_k)\ . \end{aligned}$$

Hence $\sum\{CPC(BC_j),CPC(BC_k)\}\ge CPC(BC_i)$.

[Property 5] Disjoint components aditivity

$$(for \forall BC_i,\ \forall BC_j,\ \forall BC_k \ \ such\ that\ \exists\ \forall C_i \in BC_j\ \ and\ \forall C_j \in BC_k)$$
$$BC_i = BC_j \cup BC_k\ \ and\ CC(C_i, C_j) = 0\ \ \rightarrow \sum\{CPC(BC_j), CPC(BC_k)\} = CPC(BC_i)$$

Proof: Assume that $BC_i = BC_j \cup BC_k$ and $CC(BC_j, BC_k) = 0$.

$$\sum_{C_i \in BC_j,\ C_j \notin BC_j \cup BC_k} CC(C_i, C_j) = \sum_{C_i \in BC_j,\ C_j \notin BC_j} CC(C_i, C_j)$$

and
$$\sum_{C_i \in BC_k,\ C_j \notin BC_j \cup BC_k} CC(C_i, C_j) = \sum_{C_i \in BC_k,\ C_j \notin BC_j} CC(C_i, C_j)$$

So,
$$\sum_{C_i \in BC_j,\ C_j \notin BC_j \cup BC_k} CC(C_i, C_j) + \sum_{C_i \in BC_k,\ C_j \notin BC_j \cup BC_k} CC(C_i, C_j)$$
$$= \sum_{C_i \in BC_j,\ C_j \notin BC_j} CC(C_i, C_j) + \sum_{C_i \in BC_k,\ C_j \notin BC_k} CC(C_i, C_j)$$

Hence $CPC(BC_i) = CPC(BC_j) + CPC(BC_k)$.

4 A Comparison of Proposed Metric with Existing Metrics

We have selected an online book ordering system[10] to apply the proposed component coupling metric, analyze the conventional metrics and the proposed coupling metric in this paper, and evaluate the effect of the proposed metric. The CBO[3] proposed by C.K, MPC (Message Passing Coupling)[4] proposed by Henderson-Sellers, and coupling metrics[5] proposed by Lee, existing metrics were used as the objects for comparison. We show a class diagram and derived component architecture in Fig.1, and present the number of method calls, which is analyzed by method call types between classes, to execute the function of online book ordering system in Fig. 2.

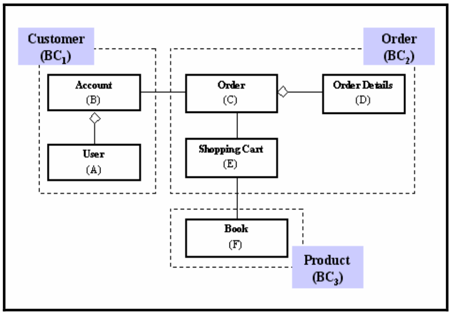


Fig. 1. Class Diagram and Components

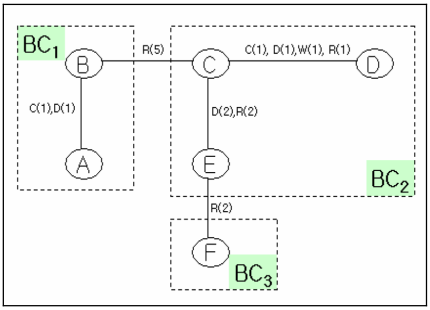


Fig. 2. Types and Number of Method calls

Fig. 3 presents the measurement results of $CPC(BC_k)$ about all possible cases of candidate components by the compounding of classes. As shown in Fig. 3, the result of proposed metric(CPC) coincides our instinct.

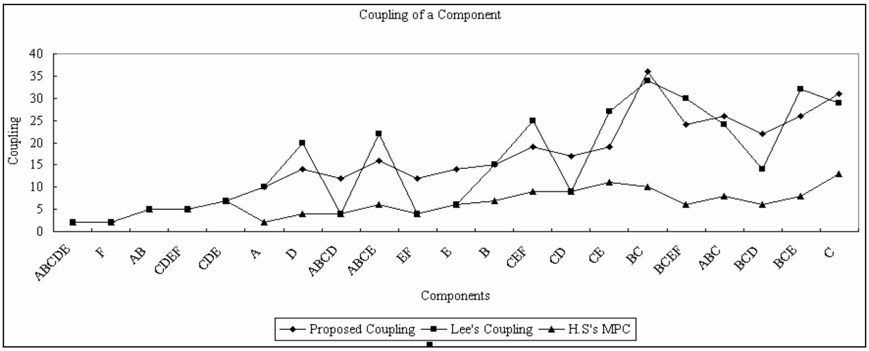


Fig. 3. The Coupling(*CPC*) of Candidate Components

Next, let a class be defined as a candidate component, and compare the coupling by $CBC(BC_l, BC_m)$ between components in Table 1. The result of Table 1 shows that the coupling between the components by H.S.'s MPC and C.K.'s CBO is the same with the number of method call. Lee presents a high coupling when aggregation and generalization relationships between components existed. In the case of the association relationship, however, the result of the coupling between components proposed by Lee presents the same value as the number of method calls. The results are not coincided with our intuition. Because, in case of conventional metrics, the types of method call between classes is not considered and, in case of object-oriented metrics, the static dependency by the structural relationship between classes is not considered. As shown in Table 1, the result of proposed metric(CBC) coincides our instinct.

Table 1. The Coupling(*CBC*) between Components

CBC(BC$_l$, BC$_m$)	**Proposed CBC**	**Lee's CBC**	**H.S's CBC**	**# of method call**
CBC(A, B)	10	10	2	2
CBC(B, C)	5	5	5	5
CBC(C, D)	14	20	4	4
CBC(C, E)	12	4	4	4
CBC(E, F)	2	2	2	2

As noted in Table 1, the candidate components of system with the lowest average coupling are (AB), (CDE), (F), and can be represented as (user, account), (Order, Order Details, Shopping Cart), and (Book) according to the coupling metric proposed in this paper. This result is coincided with our institution. Because the coupling metric proposed in this paper considers both static and dynamic dependency between components by applying the static and dynamic dependency between classes.

Table 2 shows the results of comparison between our cohesion metric and the existing metrics.

Table 2. Comparison Results between our Metric and existing Metrics

Metrics Factors	Object-Oriented Metrics	Existing Component-Based Metrics	Our Metric
Number of Method Calls	Yes	Yes	Yes
Types of Method Calls	No	No	Yes
Dynamic Dependency between Classes	Partially	Partially	Yes
Static Dependency between Classes	No	Partially	Yes
Functional Property of Component	No	Partially	Yes
Accuracy of Result by Coupling Metric	Average	Average	High

5 Conclusion Remarks

This paper proposes a component-based coupling metric, which reflects the characteristics of the component by the static and dynamic dependency applying method call types between classes, and presented results of its comparison to verify the accuracy of the proposed coupling metric. We proved the theoretical soundness of the proposed metric by the axioms of briand et al. We verified that the coupling metric proposed in this paper through the comparison with existing coupling metrics measures the designed components more precisely and quantitatively. Our coupling metric suggested in this paper can reduce software development time and efforts designing more independent components by measuring the identified components in the analysis phase or design phase. And Our maintenance efforts for software system are reduced.

References

[1] John Cheesman and John Daniels, UML Components: A Simple Process for Specifying Component-Based Software, Addison-Wesley, 2001

[2] H. Sahraoui, R. Godin, and T. Miceli, "Can Metrics Help to Bridge the Gap Between the improvement of OO Design Quality and its Automation?", In Proceedings of International Conf. on Software Maintenance, pp. 154-162, 2000.

[3] S.R. Chidamber and C.F. Kemerer, "A Metric Suite for Object-Oriented Design", IEEE Transactions on Software Engineering, vol. 17, No. 6, pp.636-638, 1994

[4] Henderson-Sellers, Brian, Object-Oriented Metrics, Prentice-Hall, 1996

[5] Jong Kook Lee, Seung Jae Jung and Soo Dong Kim, "Component Identification Method with Coupling and Cohesion", Proceedings of Asia Pacific Software Engineering Conference, pp.79 ~ 88, 2001.

[6] Byung-Sun Koh, Jae-Nyun Park, “Improvement of Component Design using Component Metrics”, KISS Journal, pp. 980-990, 2004.

[7] L.C. Briand, S. Morasca, and V.R. Basili, "Property-based software engineering measurement", IEEE Trans. Software Eng., vol. 22, no.1, pp.68-86

[8] David C. Kung, Jerry. Gao, Pei Hsia, F. Wem, Y. Toyoshima and C. Chen, "Change Impact Identification in Object Oriented Software Maintenance", Proceedings International Technical Conference on Ciecuit/Systems, Computers and Communications, 1999.

[9] David C. Kung, Jerry Gao and Pei Hsia, "Class Firewall, Test Order, and Regression Testing of Object-Oriented Programs", Journal of Object-Oriented Programming, pp. 51-65, 1995.

[10] Doug Rosenberg, Kendall Scott, "Applying Use Case Driven Object Modeling with UML", Addison-Wesley, 2001

Software Process Improvement Environment

Haeng-Kon Kim[1] and Hae-Sool Yang[2]

[1] Dept. of Computer Information & Communication Engineering,
Catholic University of Deagu, Korea
hangkon@cu.ac.kr
[2] Graduate School of Venture, HoSeo Univ. Bae-Bang myon, A-San,
Chung-Nam, 336-795, Korea
hsyang@office.hoseoa.ac.kr

Abstract. Today, in accordance with the bigger complexity of the system, the security concerns of a system have increased rapidly. One of the main concerns of the high security information system is the security evaluation to define the security function. Security evaluation of information security system is broadly used with respect to Common Criteria (CC) as ISO standards (ISO/IEC 15408:1999). The standardization of process assessment results is a key point for solving the problems. This paper suggests XML-based approach to introduce the establishment of compatible environments for process improvement on the Web. In this paper, we focus on creating SPIE (Software Process Improvement Environments) for wide acceptance of process improvement on the Web. SPIE DTD was defined to satisfy assessment output requirements in the ISO/IEC 15504. SPIE provides interoperability between applications that exchange process assessment results in machine-understandable XML format on the Web.

1 Introduction and Motivation

As CC(Common Criteria) presents common requirements about the security function and assurance means of the system, it makes enable mutual comparison among the evaluation results of the security system that performs independently. CC Evaluation Assurance Levels (EALs) have seven levels, from EAL1 to EAL7. The higher EAL levels (5, 6, and 7) are sometimes referred to as the high-assurance levels. These levels require some application of formal methods to demonstrate that the appropriate level of assurance has been met.

Researchers as well as practitioners agree to that the quality of a software product relies highly on the quality of the software process. Therefore, recent research efforts in software engineering focus on Software Process Assessment and Improvement (SPAI). Numerous approaches like the CMM, Bootstrap, SPICE, AMI were developed. Currently, ISO/IEC 15504 (SPICE) provides a framework for the assessment of software processes and has been progressed as a full International Standard . SPICE is a major international initiative to support the development of an International Standard for Software Process Assessment. The methods of Software Process Assessment

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 976 – 984, 2006.

are coming more generally into use in the management of software development, acquisition and utilization, in the face of substantial evidence of the success of such methods in driving improvements in both quality and productivity. At the same time, there has always been recognition that process assessment can be a strong and effective driver for process improvement. Much empirical evidence has accumulated demonstrating the benefits that can be derived from an assessment-based software process improvement[1,2,3].

Especially software process improvement is based on process assessment results and its effective measurement. Large companies are widely deploying Web-based software process support and improvement environments. These environments [4] will support people from physically distributed locations who collaborate in the Software Process Improvement project of the specific organization on the Web. Process assessment results have been recorded in ad-hoc manner. This makes the problem that process improvement support environments are not compatible. The standardization of process assessment results is a key point for solving this problem.

In this paper, we focus on creating SPIE(Software Process Improvement Environments) for wide acceptation of process improvement environments on the Web .

2 Background

2.1 Overview of ISO/IEC 15443

As mentioned earlier, the objective of ISO/IEC 15443 is to present a variety of assurance methods, and to guide the IT Security Professional in the selection of an appropriate assurance method to achieve confidence[5,6].

In pursuit of this objective, ISO/IEC 15443 comprises the following:

a framework model to position existing assurance methods and to show their relationships;
a collection of assurance methods, their description and reference;
a collection of assurance elements which may be part of such methods or which may individually contribute to assurance;
a presentation of common and unique properties specific to assurance methods and elements;
qualitative, and where possible, quantitative comparison of existing assurance methods and elements;
identification of assurance schemes currently associated with assurance methods;
a description of relationships between the different assurance methods and elements; and
a guidance to the application, composition and recognition of assurance methods.

ISO/IEC 15443 is organized in three parts to address the assurance approach, analysis, and relationships as follows:

Part 1, Overview and Framework, provides an overview of the fundamental concepts and general description of the assurance methods and elements. This material is aimed at understanding Part 2 and Part 3 of ISO/IEC 15443. Part 1 targets IT security managers and others responsible for developing a security assurance program, determining the assurance of their deliverable, entering an assurance assessment audit (e.g. ISO 9000, SSE-CMM, ISO/IEC 15408), or other assurance activities.

Part 2, Assurance Methods, describes a variety of assurance methods and approaches and relates them to the assurance framework model of Part 1. The emphasis is to identify qualitative properties of the assurance methods and elements that contribute to assurance, and where possible, to define assurance ratings. This material is catering to an IT security professional for the understanding of how to obtain assurance in a given life cycle stage of product or service.

Part 3, Analysis of Assurance Methods, analyses the various assurance methods with respect to relationships and equivalency, effectiveness and required resources. This analysis may form the basis for determining assurance approaches and making trade-offs among the various factors for given security applications. The material in this part targets the IT security professional who must select assurance methods and approaches.

2.2 Overview of Common Criteria

The multipart standard ISO/IEC 15408 defines criteria, which for historical and continuity purposes are referred to herein as the Common Criteria (CC), to be used as the basis for evaluation of security properties of IT products and systems. By establishing such a common criteria base, the results of an IT security evaluation will]be meaningful to a wider audience[7].

The CC will permit comparability between the results of independent security evaluations. It does so by providing a common set of requirements for the security functions of IT products and systems and for assurance measures applied to them during a security evaluation. The evaluation process establishes a level of confidence that the security functions of such products and systems and the assurance measures applied to them meet these requirements. The evaluation results may help consumers to determine whether the IT product or system is secure enough for their intended application and whether the security risks implicit in its use are tolerable.

The CC is presented as a set of distinct but related parts as identified below.

Part 1, Introduction and general model, is the introduction to the CC. It defines general concepts and principles of IT security evaluation and presents a general model of evaluation. Part 1 also presents constructs for expressing IT security objectives, for selecting and defining IT security requirements, and for writing high-level specifications for products and systems. In addition, the usefulness of each part of the CC is described in terms of each of the target audiences.

Part 2, Security functional requirements, establishes a set of functional components as a standard way of expressing the functional requirements for TOEs (Target of Evaluations). Part 2 catalogues the set of functional components, families, and classes.

Part 3, Security assurance requirements, establishes a set of assurance components as a standard way of expressing the assurance requirements for TOEs. Part 3 catalogues the set of assurance components, families and classes. Part 3 also defines evaluation criteria for PPs (Protection Profiles) and STs (Security Targets) and presents evaluation assurance levels that define the predefined CC scale for rating assurance for TOEs, which is called the Evaluation Assurance Levels (EALs).

In support of the three parts of the CC listed above, it is anticipated that other types of documents will be published, including technical rationale material and guidance documents.

2.3 XML Application

XML (eXtensible Markup Language) is a simplified subset of SGML that is intended to use on the Web [8]. XML is more powerful than HTML because it is extensible. Users can define new tags, attributes, and are not limited to the finite set that never seems to satisfy anyone. XML has many advantages. The first one is that XML is platform and system independent. It works as well on one computer as it does on the others. XML is designed to work with any XML software. One of other major advantage is that we can create our own tags. Any XML-aware software will be able to work with our custom XML application.

2.4 Software Process Assessment and Improvement

A software process assessment is a disciplined examination of the software processes used by an organization, based on a process model. Its objective is to determine the maturity level of those processes, as measured against a process improvement roadmap [9]. The result should identify and characterize current practices, identifying areas of strengths and weaknesses, and characterize current practices to control or avoid significant causes of poor quality, cost and schedule. The assessment findings can also be used as indicators of the capability of those processes to achieve the quality, cost and schedule goals of software development with a high degree of predictability [6].

Currently, a number of software process improvement roadmaps are publicly available. The most notable are the Capability Maturity Model (CMM) [10] , ISO 9001 with its associated guide ISO 9000-3, and the ISO/IEC 15504(SPICE).

2.5 SPICE (Software Process Improvement and Capability dEtermination)

SPICE is a major international initiative to support the development of an International Standard for Software Process Assessment. The project has a main goal to develop a working draft to satisfy a standard of software process assessment. This goal of the project was achieved in June 1995, with the release of Version 1 of a draft standard for software process assessment (the SPICE Documents) to WG10 for international ballot among the standards community, following the normal process for development of international standards. Following this ballot, the documents have been carried through the international standardization process and have been published as ISO/IEC TR 15504:1998 - Software Process Assessment. WG10 has now commenced work to revise the TR with a view to ultimate publication as a full International Standard. ISO IEC TR 15504 has also been validated internationally in the SPICE trials [8] where it has proven useful for performing assessments.

SPICE defines a two dimensional reference model for describing processes and capability used in a process assessment. A reference model defines a set of processes and a framework for evaluating the capability of the processes through assessment of process attributes structured into capability levels [11]. Process dimension is characterized by the process purposes that are the essential measurable objectives of a process, and the expected outcome of the process that indicates its successful completion. Process capability dimension is characterized by a series of process attributes applicable to any process, and represents measurable characteristics necessary to manage a process and to improve its performing capability. A capability level is a set of attributes that work together to provide a major enhancement in the capability to perform a process. SPICE specifies six capability levels in the reference model (numbered 0 to 5). The capability levels incorporate nine process attributes.

SPICE provides guidance when using software process assessment as part of a framework and method for performing software process improvement in a continuous manner. The overall context of process improvement is shown in figure 1.

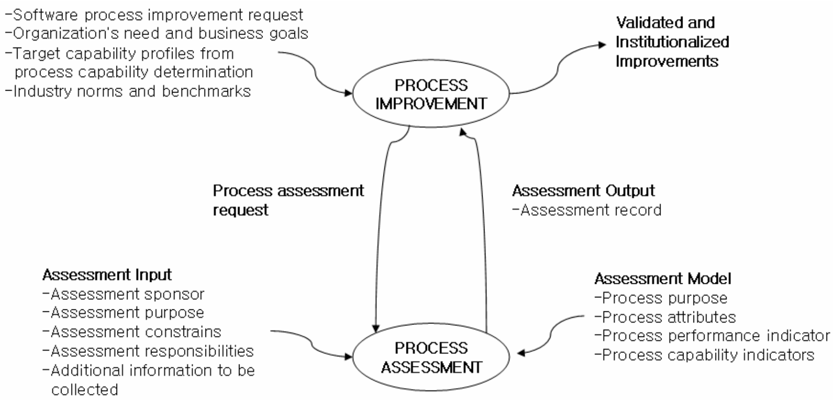

-Software process improvement request
-Organization's need and business goals
-Target capability profiles from process capability determination
-Industry norms and benchmarks
PROCESS IMPROVEMENT
Validated and Institutionalized Improvements
Process assessment request
Assessment Output
-Assessment record
Assessment Input
-Assessment sponsor
-Assessment purpose
-Assessment constrains
-Assessment responsibilities
-Additional information to be collected
PROCESS ASSESSMENT
Assessment Model
-Process purpose
-Process attributes
-Process performance indicator
-Process capability indicators

Fig. 1. Context of process improvement

3 Design of Software Process improvement Environments

3.1 Overview

SPIE(Software Process improvement Environments) is an application of XML used to include process assessment result proposed by SPICE. The goal of SPIE is to annotate and augment standard process assessment result and to deliver the essential information for SPI on the Web in an interchangeable text format that is easier to interpret. SPIE provides interoperability between applications that exchange process assessment results in machine-understandable format on the Web. SPIE emphasizes facilities to enable automated processing of process improvement. If Web-based process improvement environments support SPIE, they can interchange a SPIE file including its process assessment results, and import it to analyze process assessment results, deriving action plan automatically. The results of an assessment conducted by SPICE formally comprise a set of process attribute ratings for each process in the assessment scope. The set of process attribute ratings is termed the process profile. Additional information on the context of the assessment and the processes assessed must also be recorded as part of the assessment record because requirements for recording the assessment outputs are contained in ISO/IEC 15504-3. Attribute ratings may be used to calculate a capability level rating for the assessed process. Whatever the final format of the process output, it is essential to provide the clear traceablility to the processes and process attributes contained in the reference model to enable the process of calculation to be verified. SPIE framework was designed to satisfy this kind of constraint with the completeness of process assessment results for process improvement.

It should be noted that a SPIE file contains full details of the process context in the assessment record. This file will also include additional information collected as part of the assessment, and required as inputs to the process improvement or process capability determination activities to follow on from assessment. In order to enable process improvement using SPIE, process improvement planner must follow a structured approach to convert the software process findings of a SPIE file into an improvement action plan. Developing an improvement action plan out of a SPIE file could be achieved in a series of steps as follows:

1. Converting the assessment findings in a SPIE file into recommendations (manually or automatically)
2. Converting the recommendations into action plan
3. Grouping the actions into action plan/work package.
4. Allocating the action plans to the software process improvement teams

3.2 Structure of SPIE

SPIE was invented for exchanging descriptions of process assessment results proposed by SPICE between people and computers within the intranets or on the Web.

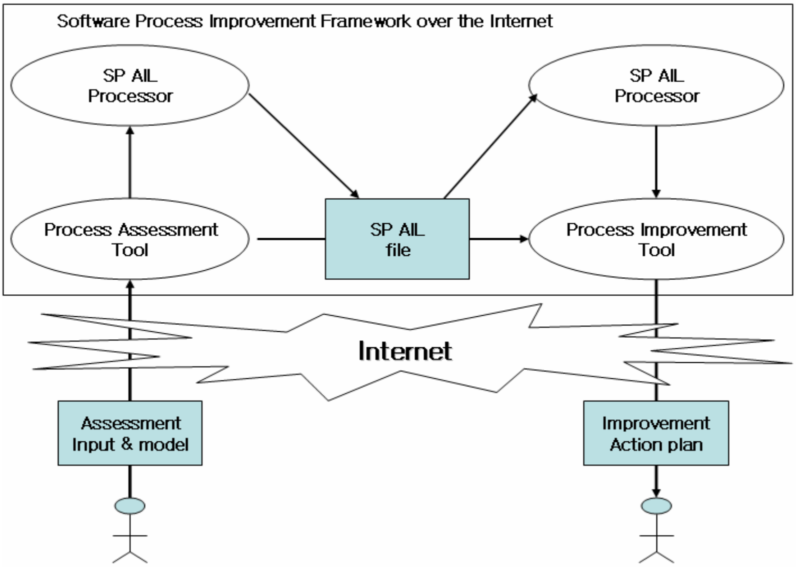


Fig. 2. Process improvement environment using SPIE on the Web

All SPIE elements are contained in one of following two categories:

1) Assessment Input Information
Assessment input information elements describe assessment input structure contained in SPICE. The assessment input defines the purpose of the assessment, the scope of the assessment, and what constraints, if any, apply to the assessment. The assessment input also defines the responsibilities for carrying out the assessment. For example, the <SCOPE> element represents the scope of the assessment. Each element corresponds to an element of the assessment input specification in SPICE.

2) Assessment Output Information
In addition, assessment output information elements describe assessment output structure contained in SPICE. Assessment output includes a set of process profiles and capability level rating for each process assessed. For example, the <ProcessProfile> element represents each process profiles produced by SPICE process assessment. Information collected during the assessment, in particular the capability level and process attribute ratings, will be analyzed to identify areas for improvement and derive action plan, and integrate it with the process improvement program plan. Thus, the <ProcessProfile> element is very important because it contains the core information required for deriving action plan from assessment output.

4 Example

There are several scenarios where an XML representation of SPIE definitions is useful. The SPIE format can be used as an intermediate format on the Web between a SPIE compiler/parser and post processing tools such as process improvement environments in Figure 2.

The SPIE format can be used with XSLT post processors to generate various documentations related to process improvement in various formats.

The XML format makes it possible to access a SPIE file easily from a variety of programming languages. XML parsers are available in Java, C, C++, Tcl, Perl, Python, and GNU Emacs Lisp in both commercial and open source forms.

```
    <!ELEMENT PA1.1Rating (#PCDATA)>
    <!ELEMENT PA2.1Rating (#PCDATA)>
    <!ELEMENT PA2.2Rating (#PCDATA)>
    <!ELEMENT PA3.1Rating (#PCDATA)>
    <!ELEMENT PA3.2Rating (#PCDATA)>
    <!ELEMENT PA4.1Rating (#PCDATA)>
    <!ELEMENT PA4.2Rating (#PCDATA)>
    <!ELEMENT PA5.1Rating (#PCDATA)>
    <!ELEMENT PA5.2Rating (#PCDATA)>
    <!ELEMENT AssessmentEvidence (BasicEvidence, ExtendedEvidence?)>
    <!ELEMENT ExtendedEvidence ANY>
    <!ELEMENT BasicEvidence (PA1.1Indicators, PA2.2Indicators, PA3.1Indicators,
PA3.2Indicators, PA4.1Indicators, PA4.2Indicators, PA5.1Indicators,
PA5.2Indicators)>
```

```
<!ELEMENT PA1.1Indicators (Indicator+)>
<!ELEMENT PA2.1Indicators (Indicator+)>
<!ELEMENT PA2.2Indicators (Indicator+)>
<!ELEMENT PA3.1Indicators (Indicator+)>
<!ELEMENT PA3.2Indicators (Indicator+)>
<!ELEMENT PA4.1Indicators (Indicator+)>
<!ELEMENT PA4.2Indicators (Indicator+)>
<!ELEMENT PA5.1Indicators (Indicator+)>
<!ELEMENT PA5.2Indicators (Indicator+)>
<!ELEMENT Indicator (Assessors?, Assessees?, Workproduct?, Question, Answer, Findings?, Note*)>
<!ELEMENT Workproduct (#PCDATA)>
<!ELEMENT Question (#PCDATA)>
<!ELEMENT Answer (#PCDATA)>
<!ELEMENT Findings (#PCDATA)>
<!ELEMENT Strength (#PCDATA)>
<!ELEMENT Weakness (#PCDATA)>
<!ELEMENT ProcessCapabilityLevel (#PCDATA)>
```

5 Conclusion

This paper suggests a basic mechanism for compatible framework of software process improvement environment on the Web. This type of environment is hardly used in industrial practice because introducing this type of tool support usually involves the standardization of assessment output data for tool vendors and process performers.

This paper suggests a XML-based approach to solve it. SPIE DTD was defined to satisfy assessment output requirements in the ISO/IEC 15504. The paper concludes with some examples of SPIE and leaves future work with their development and usage.

Acknowledgement

This research work has been supported by the IT Research Center (ITRC) Project in 2005.

References

1. T. Rout, "SPICE: A Framework for Software Process Assessment", Software Process Improvement and Practice Journal, Pliot Issue, pages 57-66, August 1995.
2. F. Cattaneo, A. Fuggetta, D. Sciuto, "Pursuing coherence in software process assessment and improvement", Software Process: Improvement and Practice", Volume 6, Issue 1, 2001.
3. Campbell, M., Tool Support for Software Process Improvement and Capability Determination: Changing the Paradigm of Assessment. Software Process Newsletter 4, p12-15, 1995.

4. G.A.Bocler and R. N. Talylor, “Endeavors: A Process System Integration Infrastructure”, Proceedings of the Fourth International Conference on the Software Process, Brighton, England, December 1996.
5. Howard Rubin, "Software Process Maturity", Computer Channel Inc, 1993
6. Sami Zahran, Software Process Improvement (Addision-wesley, 1998).
7. Paulk, M.C, "The Evolution of the SEI's Capability Maturity Model for Software", Software Process: Improvement and Practice, Pilot issue, pp3-15, 1995.
8. F. Maclennan, G. Ostrolenk. The SPICE Trials: Validating the Framework. Proceedings of 2nd International SPICE Symposium, 1995.
9. ISO/IEC 15504 TR2:1998, Part 2: A reference model for processes and process capability, ISO/IEC JTC1/SC7, 1998.
10. W3 Consortium, "W3C Recommendation: Extensible Markup Language 1.0", "http://www.w3.org/TR/1998/REC-xml-19980210", 1998.
11. Rombach, H.D. & Verlage, M., "Directions in Software Process Research", Advances in Computers, Vol. 41. pp1-63, 1995.

A Design Technique of CBD Meta-model Based on Graph Theory

Eun Sook Cho[1], So Yeon Min[1], and Chul Jin Kim[2]

[1] Dept. of Software, Seoil College,
49-3 Myeonmok-8 Dong, Jungnang-Gu, Seoul 131-702, Korea.
escho@seoil.ac.kr, symin@seoil.ac.kr
[2] Digital Solution Center, Samsung Electronics Co.,
Union Steel Bldg. 890, Daechi4-Dong, Gangnam-gu, Seoul 135-534, Korea
chuljin777.kim@samsung.com

Abstract. There are several component reference models for component development. However, there is few integrated and generic reference model among reference models. That results in the problem of interoperability among component designs. In this paper, we propose an integrated component meta-model to support consistency and interoperability between component designs. Also we validate a proposed meta-model through graph theory. We expect that new meta-model will be added and extended because proposed meta-model is represented with UML's class diagram.

1 Introduction

Currently interests of component-based software development are being increased. For example, there are CBD96, RUP, Catalysis, Advisor, Fusion, and so on. Also, there are several CASE tools like as Rose, Together, and COOL series and technology platforms such as EJB, CCM, .NET and so on.[1]. These various methods, tools, and platforms have not a standard reference model, but a unique model. Each component reference model provides different notations and modeling elements for the same concept. A few reference models reflect characteristics of components fully on its meta-model. This raises the problems of inconsistency and low interoperability between components developed by different component reference model. Also, different reference models increase difficulties of communication between component designers and developers. In order to address the problems, we suggest a generic reference model, which is integrated and unified several reference models, as a forms of meta-model based on UML's class diagram [8].

The structure of this paper is as follows. Section 2 reviews existing component reference models as related approaches. Section 3 describes a generic and unified component reference model which integrates existing component reference models and suggests specification level and implementation level. Section 4 introduces graph grammar of component to verify proposed reference model, and validates proposed reference model based on proposed graph. Finally, concluding remarks and future works are described in Section 5.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 985 – 993, 2006.

2 Limitations of Existing Researches

2.1 SEI's Component Reference Model

SEI defines a component as a software implementation executable in physical and logical device. Therefore, a component implements one or more interfaces [2]. This reflects that a component has a contract. Components developed independently depend on specific rules and different components can be interoperable with standard methods. Also, components can be executable dynamically in run time. Component-based system is a system developed based on independent component types executes specific roles in a system. Types of each component are represented with each interface.

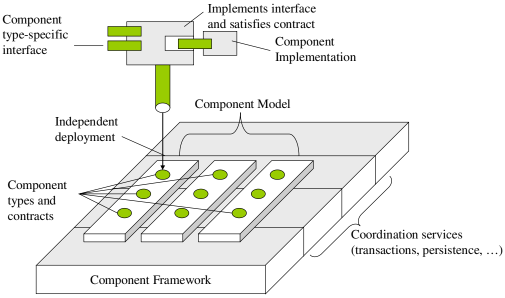

Component type-specific interface
Implements interface and satisfies contract
Component Implementation
Component Model
Independent deployment
Component types and contracts
Coordination services (transactions, persistence, ...)
Component Framework

Fig. 1. Component Reference Model of SEI

A component model defines a set of specifications of component types, interfaces, and interaction pattern between component types. And component model is represented as a specification of standard and contract for component developer. Dependency of specific component model is a property that distinguishes one component from other components.

2.2 Perrone's Component Reference Model

Perrone[3] defines a component as an unit which consists one or more classes and provides functions of classes with interfaces as depicted in Figure 2. In this research, a component is described as the concept of larger unit than the concept of existing class. Also, a component is described a unit which encapsulates separated problem domain.

Figure 2 represents basic elements contained in a component model. In this figure, Perrone defines that a component model is a component itself. Also, Perrone defines that a container is an environment in which a component is operated or worked. A container provides services that components require to send or receive messages in a standard way.

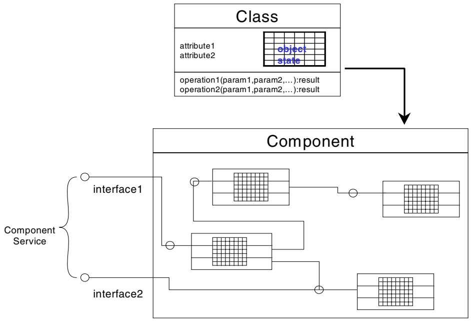


Fig. 2. Perrone's Component Reference Model

2.3 CORBA Component Reference Model

CORBA Component Model(CCM) separates a component into two-phases[6]. The one is basic component, and the other is extended component. Basic component provides a mechanism componentized CORBA object and can be mapped or integrated into EJB component. The extended component provides many functions than basic component. Both basic component and extended component are managed by component home.

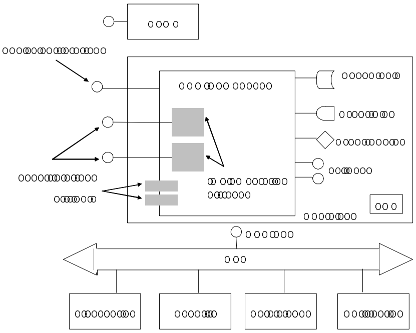


Fig. 3. CCM 's Reference Model

2.4 EJB's Component Reference Model

EJB component model is similar to CCM component model[7]. A component is executed in container and managed by home object. However, besides CCM component, EJB component is referenced by one component interface. Therefore, a bean does not have several interfaces. EJB component model is separated into local or remote interface and internal implementation logic.

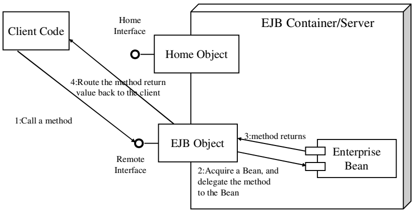


Fig. 4. EJB Reference Model

3 Generic Component Reference Model

In this section, we suggest a new component reference model based on component reference model of section 2. Figure 5 is a generic component reference model. Figure 5 describes both structural elements and dynamic elements of a component. Dynamically component workflows are occurred through calling operations of provide interface in a component depicted in Figure 5.

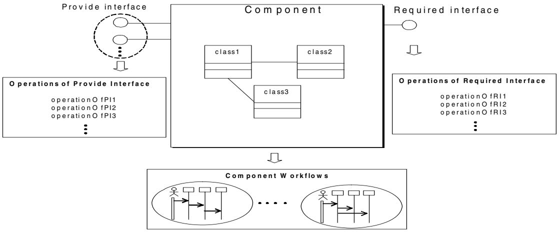


Fig. 5. Generic Component Reference Model

Figure 6 describes meta-model of static elements of a component with UML's class diagram[8].

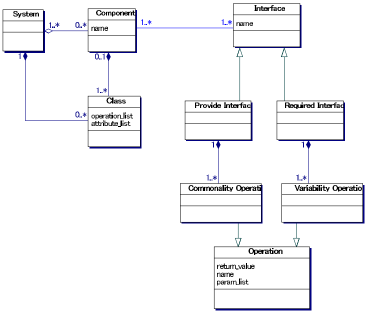


Fig. 6. Meta-model of Static Elements in a Component

3.1 Meta-model of Specification Level

Specification level meta-model describes common information among CBD methodologies. Minimal meta-model of specification level represents component definition commonly contained in all of CBD methodologies.

A component consists of component declaration and component definition. Component declaration contains one or more interfaces containing one or more method declarations. Interface is classified into provided interface and required interface. Method declaration only contains method's signature, zero or more pre-conditions and post-conditions. There are one or more classes in component definition. These classes implement interfaces declared in component declaration. Definition of class and attribute is based on UML's definition. Also, component specification contains one or more interface specifications and components.

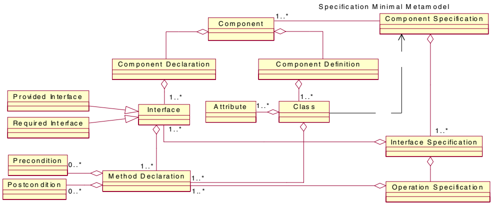


Fig. 7. Minimal Meta-model of Specification Level

Maximal meta-model of specification level is depicted in Figure 8. Maximal meta-model represents information defined in CBD methodologies.

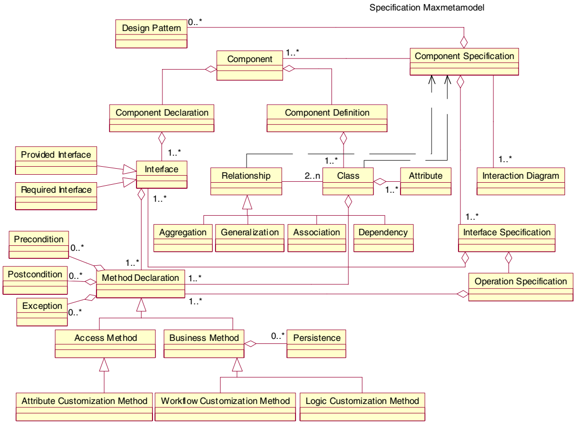


Fig. 8. Maximal Meta-model of Specification Level

In the specification maximal meta-model, method declaration is classified into access method and business method. Attribute customization method is a subclass of access method, because attribute customization method contains methods for variant attribute type as well as variant attribute values while access method contains get or set methods.

3.2 Meta-model of Implementation Level

Implementation level meta-model is used to implement components using specific component platforms such as EJB, CCM, and COM. Meta-model of implementation is divided into two types; minimal meta-model(Figure 9) and maximal meta-model(Figure 10).

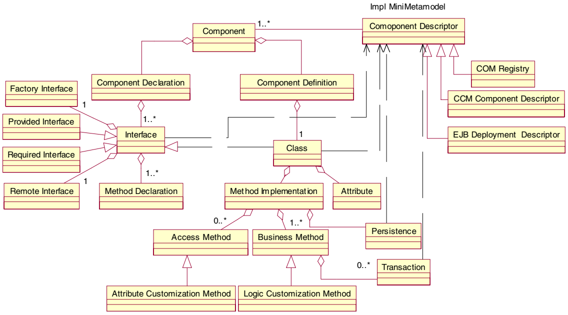


Fig. 9. Minimal Meta-model of Implementation Level

Minimal meta-model of implementation level represents additional information related with component implementation. For example, information related with transaction is reflected on business method. While component information is described in component specification in specification level meta-model, it is described in component descriptor in implementation level meta-model. Maximal meta-model of implementation level describes comprehensive of component platforms. Therefore, there is additional information needed in component implementation according to component platform.

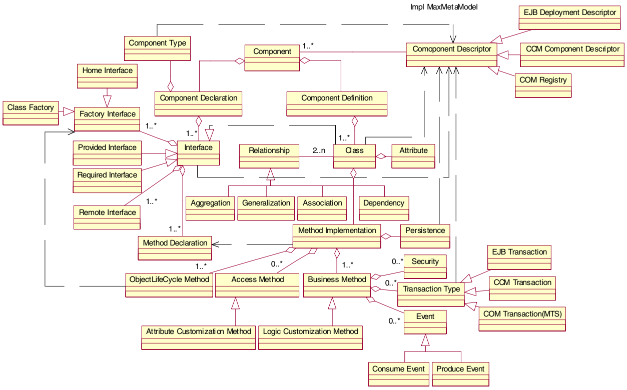


Fig. 10. Maximal Meta-model of Implementation Level

4 Verification

In this section, we validate or verify whether proposed meta-model is well defined or not by using OMG's Meta Object Facility(MOF).

4.1 Verification of Meta-model Through MOF

MOF defines meta meta-model required composing, validating, and transforming expressible all of meta models including UML meta model.[10]. In order to verify whether component meta-model is well defined or not, we first should prove that proposed component meta-model is instance of MOF. And then, we should prove that proposed meta-model conforms to rules of MOF. The fact of component meta-model confirming to MOF can be proven by the relationship between MOF and component meta-model.

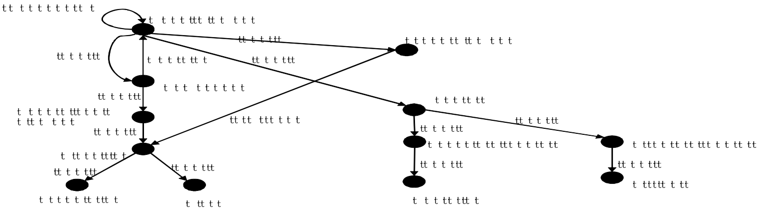


Fig. 11. BMOF Graph

MOF basic elements can be regarded as a graph including node and arc. A node represents an element of MOF, and arc means the relationship between elements. Therefore, MOF might be transformed into BMOF graph such like Figure 11. Finally, we also can transform minimal meta-model of specification level into a graph such like Figure 12.

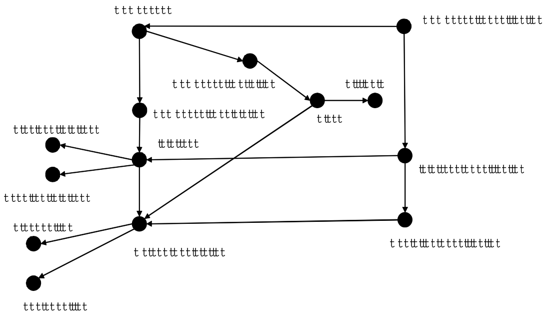


Fig. 12. Graph for Minimal Meta-Model of Specification Level

A graph can be described a pair of node and arc. Therefore, BMOF is represented as follows:

BMOF = < N_b, A_b>
N_b = {Model Element, Namespace, GeneralizableElement, Package, Classifier, Association, Class, Typed Element, Parameter, Structural Feature, Feature, Behavioral Feature, Operation, Attribute}
A_b = { Depends On, Contains, … }
Minimal meta-model of specification level, **SMGraph**, is represented as follows:
SMGraph = <N_s,A_s>
N_s = {Component, Component Specification, Component Declaration, Component Definition, Interface Specification, Operation Specification, Class, Attribute, Interface, Provided Interface, Required Interface, Precondition, Postcondition, Method Declaration}
A_s = {$A_{component\_definition}$, .. }
BMOF' becomes a basis to check the consistency of **SMGraph**. In order to have consistency, there should be nodes of **BMOF**' mapped into all nodes of **SMGraph**. Also, if there is an arc relating two nodes for any two nodes in **SMGraph**, there should be an arc for two nodes of **BMOF**'. Expression of those is as follows:

F : first(SMGraph) →first(first(BMOF'))
$\forall n_i, n_j\ (n_i \neq n_j) \in$ dom F, arc(n_i, n_j) $\in$ second(SMGraph).
$\exists$arc(F (n_i), F (n_j))$\in$ rand F

In order to prove the consistency of **SMGraph,** the function from **SMGraph** into **BMOF'** should be defined. Function **F** is defined as follows:

F(component)= Classifier ....(1)
F(Component Specification)= Package....................(2)
F(Component Declaration)= Package......................(3)
F(Component Definition)= Package.........................(4)
F(Interface)= Classifier.......................................(5)
F(Provided Interface)= Classifier............................(6)
F(Required Interface)= Classifier............................(7)
F(Attribute)= Attribute...(8)
F(Class)= Class...(9)
F(Method Declaration)= Operation.........................(10)
F(Precondition)=Constraint...................................(11)
F(Postcondition)= Constraint.................................(12)
F(Interface Specification) = Package......................(13)
F(Operation Specification) = Package.....................(14)

All arcs of **SMGraph** are mapped into all arcs of **BMOF**'. It means that all nodes of **BMOF**' are kinds of **Model Element**, and there are relationship of *"Depends On"* between **Model Element**.

Following mappings for maximal meta-model of specification level are added.

F(Design Pattern)= Classifier................................(16)
F(Interaction Diagram)= Classfier..........................(17)
F(Exception)=Exception......................................(18)
F(Persistence)=Tag..(19)
F(Relationship)=Association.................................(20)

Following mappings for minimal meta-model of implementation level are added.
F(Component Descriptor)=Costraint………………..(21)
F(Transaction)=Tag…………………………………(22)

Maximal meta-model of implementation level includes following mappings.

F(Security)=Tag...…………………………………(23)
F(Transaction Type)=Tag…………………………(24)
F(Event)=Operation……………………………….(25)

5 Conclusion Remarks

We define and propose meta-models for component with respect to generic view, specification view, and implementation view. Also, each meta-model of specification level and implementation level is divided into minimal meta-model and maximal meta-model. In order to verify the correctness and soundness of proposed meta-models, we use graph theory and MOF.

We expect that models of various methodologies and platforms will be integrated as well as new model elements are added or extended easily by applying proposed meta-model.

References

1. Heineman, G. T., Council, W. T., Component-based Software Engineering, Addison Wesley, 2001.
2. Bachman, F., et. al., “Volume 2, Technical Concepts of Component-based Software Engineering”, Carnegie Mellon Software Engineering Institute, 2000.
3. perrone, p., Building Java Enterprise Systems with J2EE, Sams Publishing, 2000.
4. Butler Group, “Catalysis: Enterprise Components with UML”, at URL: http://www. catalysis.org, pp.2, 1999.
5. Desmon F. D’Souza and Alan Cameron Wills, Objects, Component and Frameworks with UML, Addison Wesely, 1999.
6. OMG, Final FTF Report of the Component December 2000, OMG Inc., 2001.
7. Roman, E., Mastering Enterprise JavaBeans, Jon Wiley and Sons, Inc., 2002.
8. UML Specification v1.4, OMG,Inc., September, 2001.
9. D. Harel, A. Naamad, “The STATEMATE semantics of Statecharts,” ACM Transactions on Software Engineering and Methodology, Vol.5, No.4, pp.293-333, 1996.
10. Meta Object Facility Specification, OMG, URL: http://www.omg.org.
11. Akehurst, D.H, “Model Translation: A UML-based specification techniques and active implementation approach”, PhD Thesis, University of Kent at Canterbury, 2000.

Description Technique for Component Composition Focusing on Black-Box View

J.H. Lee and Dan Lee

Software Technology Institute, Information and Communications University,
Seoul, Korea
{puduli, danlee}@icu.ac.kr

Abstract. As component-based software is developed by integrating components that are implemented independently, expressing the usage protocols of each component is essential. However, there is no known proper way to describe them comprehensibly from the point of component user or developer. Black-box (external) point of view of component composition sees component-based development from the user's or the system assembler's point of view. But a description technique necessary to specify the dynamic constraint explicitly is necessary to define the external view more precisely. The key contribution of this paper is to present a technique for describing the structure of components in black-box view using UML 2.0. First, we present the relevant UML notations for describing the black-box point of view and then provide diagrams showing their usage. We further illustrate how this leads to a component based software specification of the structure of composition focusing on the black-box view.

1 Introduction

Even though there are some commonalities among classes, components and subsystems, they are used unclearly due to the non-obvious subtle differences among them in UML 1.x. For example, does a subsystem mean just a big component? In that case, how big a component should it be to be qualified as a subsystem? These kinds of questions have some vague problems that are difficult to give a clear answer. UML 2.0, on the other hand, defines components as the special case of the more general concept of a structured class and subsystems as the special case of the component concept. Therefore, a choice among them will be determined by the basis of objective criteria [1].

In the same manner, relationships between components through contracts must be specified along with the operations that the interfaces provide; the component diagram in UML 1.x provides only a basic binding concept between interfaces which make it difficult to describe component usage relationships. To facilitate a large scale system modeling, UML 2.0 extends some existing elements such as activities, interactions, and state machines and introduces new elements such as component structures. For example, it is possible for a sequence diagram to include some other sequences while the protocol state machine can be attached to the component structure diagram. Owing to these extensions, behavior description of a component

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 994 – 1002, 2006.

can be reused in the same or other component description [1]. As a result, it becomes possible to describe behavioral patterns that can be applied in different contexts. That is, we can reuse same specification in multiple places within the same view (e.g. dynamic view, component view, etc.) or some other views.

In this paper, we suggest some alternative ways to describe the behavioral aspect description in black-box view that shows the optimal levels of information in describing the relationships between interacting components and their usages by using UML 2.0. The rest of this paper is organized as follows. In section 2, the notations for the component relationship description and the behavioral aspect description in the black-box view are explained. In section 3, a method for the black-box view description is introduced by using the notations explained in section 2. In section 4, the advantages and disadvantages of black-box point of view description and contributions of this paper are discussed. Lastly, we present the future research direction and then draw conclusions.

2 Related Works

2.1 Basic Concepts for Black-Box View Description

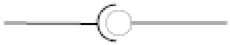
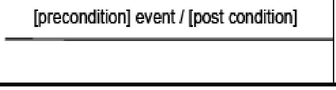
From user's(external) point of view, a component consists of a set of provided services which form the provided interface of the component. The services provided by other components are called required services, which are provided by a required interface. Black-box view as an external point of view describes a component composition and how to use it. Through adding black-box view specification to an architectural description, we can provide user or developer with the knowledge necessary to use and understand the components in a proper level of abstraction.

[2] presents a definition and specification for component software from black-box point of view. Also, [2] makes defining state machine or state chart of contracts and component possible by introducing control variables for them via applying the same method as an interface. In some cases, we can add a CSP specification describing the sequence of the interface method in contracts. However the definition and specification of [2] based on mathematical logic has formal characteristics. This mathematical logic is too difficult for designer or developer to understand and CSP is almost the same.

Now, UML 2 supports new notations for describing component composition and is upgraded to fit for specifying the order of invocation of object's operation in a convenient way by using protocol state machine. Especially, as protocol state machine doesn't preclude specific behavioral implementation and enforces legal usage scenarios of classifier, it can be associated with interface and port [3].

UML 2 provides following notations and diagrams for specifying architectural connector defined in [4], which describes it by using an additional profile borrowed from architectural description language concept such as ACME or C2SADL as a collection of protocols.

Table 1. UML 2 notations for specifying architectural connector

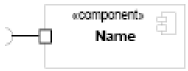
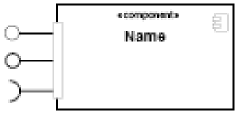
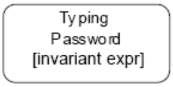
Path Type	Notation	Included in
Assembly Connector		Structure Diagram
Connector		Composite Structure Diagram
Protocol transition		Protocol State Machine Diagram

Though there are many improvements in UML for describing component and port, it is insufficient to describe connector presented in architecture views [7]. But it is possible to model component behavior from the point of black-box by attaching protocol state machine to the structure of components using assembly connector and connector notation in Table 1.

An assembly connector maps a required interface of a component to a provided interface of another component in a certain context. It specifies composition between components as a connector wiring between provided and required interfaces in a composite structure diagram of component

We use connector notation for attaching protocol state machine diagram to the external structure diagram of components. Connector notation in Table 1 specifies a link that enables communication between two or more instances. This link may be an instance of an association, or it may represent the possibility of the instances being able to communicate because their identities are known by virtue of being passed in as parameters, held in variables or slots, or because the communicating instances are the same instance [3].

Table 2. Notations for describing the structure of components on black-box view

Node Type	Notation
Component with required Port(typed by Interface)	
Component with complex Port(typed by provided and required interfaces)	
State with invariant	

Protocol transition specifies a legal transition for operation. Transitions of protocol state machines have the following information: a pre-condition (guard), on trigger, and a post-condition. Protocol state machine also can describe invariants by using state with invariant notation in Table 2. Every protocol transition is associated to zero or one operation that belongs to the context classifier of the protocol state machine [3].

Besides these notations, we use sub-state machine to describe protocol state between two connected components. Because protocol state machine only describes the context of a classifier in UML 2.0, we use sub-state machine to represent whose state is related to another classifier context.

2.2 Contract-Based Description

The principle behind the use of pre- and post condition is often referred to as the design by contract (DbC) [8]. A contract describes the services provided by an object. It describes the conditions under which the service will be provided and result of the service that is provided. In a component world, a client can interact with a given component through interface. An interface is a set of named operations that can be invoked by the client. The contract states what the client needs to do to use the interface. Particularly, because component is used as a black-box if contracts are not defined precisely, it can be difficult for clients to use the components and get correct returns. On the level of an individual operation of an interface, there are pre- and post conditions which must be satisfied. DbC is composed of these pre condition, post condition, and invariant operations basically. But a component contract can become much more complicated than DbC and there are additional contacts according to what component architecture is used. [5] insists that we have to decide how components behave before trusting components. Namely, they introduce the need of contracts in component specification and present four levels of contracts.

- Basic Contracts(IDL, Type systems)
- Behavioral Contracts(Invariants, Pre/Post conditions)
- Synchronization Contracts
- Quality-of-Service Contracts

Synchronization contracts make sure that services are atomic or executed as transactions. It specifies strategies to manage intra-component concurrency. Quality-of-Service contracts provide quality of service parameters, such as maximum response delay, average response, quality of result, and throughput.

3 Black-Box View Description

In this section, we illustrate how the notations and concepts introduced in section 2 lead to a component based software specification of the structure of composition focusing on the black-box view. To facilitate explanation we consider a basic online ordering system that is developed in several components. We analyze the problem domain as follows:

Henri wants to enter a new market by offering his products through an online ordering system on internet. Online ordering will allow customers to connect with the ordering system through the internet. The customer will be able to select from a list of items. The customer's selection will then be transferred to the server. The server will calculate the price of the items and send the price back to the client. The customer can either accept or decline. If the customer accepts, an order will be printed at the server [6].

Following diagram shows the structure of components of the example system.

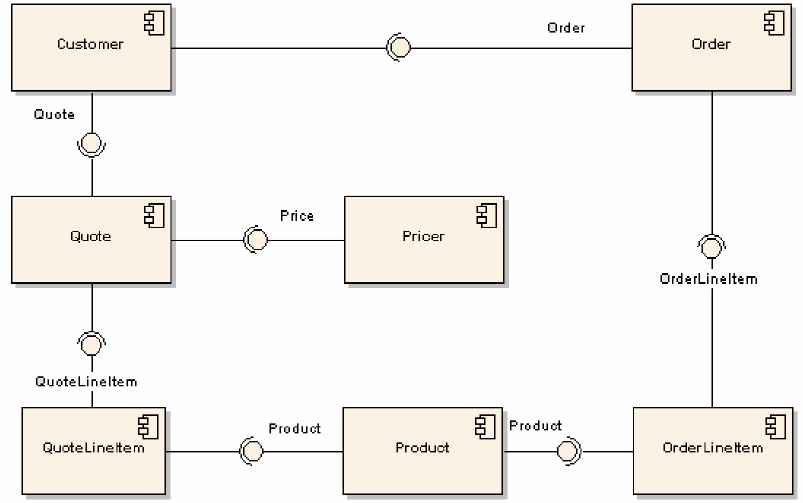


Fig. 1. The structure of components of example system

To shorten the presentation of the specification, we specify behaviors only about adding a product to quote in the *Quote* component. When we add a product to quote, the *addProduct* operation provided by *Quote* component determines whether a line-item exists already for this product. If it exists, simply increase the quantity, otherwise it makes a new quote line-item. Through the structure diagram of Fig. 1, we can see that *Quote* component has required/provided relationship with *QuoteLineItem* component. An individual line-item of *QuoteLineItem* component represents one particular product that the customer wants as well as a quantity for that product. Also, *QuoteLineItem* is in a assembly composition relation (required/provided interface) with *Product* component.

In UML 2.0, because protocol state machine package merges interface, ports, and BehaviorStateMachines package, interfaces can own a protocol state machine. Protocol state machines help define the usage mode of the operations and receptions of a classifier by specifying:

- In which context (under which states and pre conditions) they can be used
- If there is a protocol order between them
- What result is expected from their use [3]

The states of a protocol state machine (protocol states) present an external view of the class that is exposed to its clients. Depending on the context, protocol states can correspond to the internal states of the instances as expressed by behavioral state machines, or they can be different.

In our example, behavior of adding product to quote is achieved through several assembly composition relations. As several components participate in that behavior, describing black-box view for this is not easy. According to UML superstructure specification, when two ports are connected, the protocol state machine of the required interface (if defined) must be conformant to the protocol state machine of the provided interface (if defined). So, at first we specify *Quote* component's behavior associated with *QuoteLineItem* in the following figure.

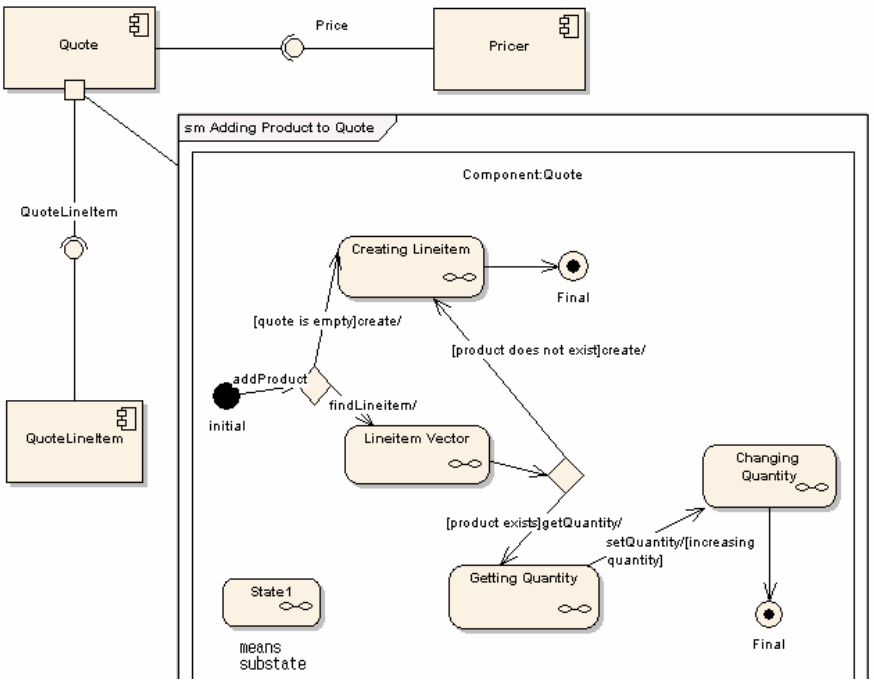


Fig. 2. The structure diagram of components attached protocol state machine. *Adding Product to Quote* state machine diagram is attached to Quote component's port inter-operating with *QuotoLineItem* component.

Fig. 2 shows that the protocol state machine for the behavior of adding product to quote which is owned by customer has been attached to *Quote* interface. In Fig. 2, we attached it to a port for representing that this behavior is one of *Quote* interface's interactions with the external environment. When *addProduct* operation is called, *create* operation will be invoked in the case of the empty state of quote (pre-condition) and *findLineitem* operation will be called in other cases. Each state indicated as substates, marked as ∞, is protocol state machine associated with provided interface and is called by required interface. *Creating Lineitem* state, which generates line-item, and *Lineitem Vector* state which contains line-items and their quantity chosen by customer, are treated as substates because both of them are related with *QuoteLineItem* component's provided interface.

After describing the behavior of a component which has required interface, we specify each substate presented in the description. Fig. 3 shows a protocol state machine diagram including substate connected to *QuoteLineItem* component's provided interface (interface of *Product* component).

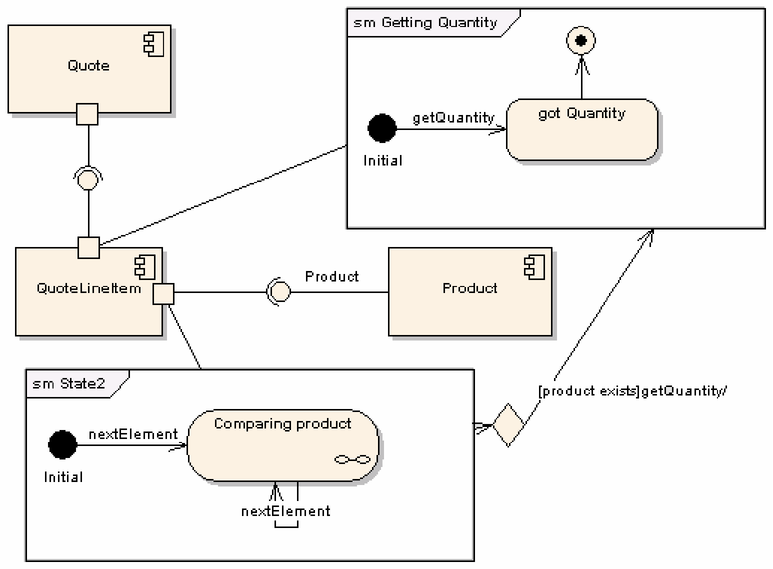


Fig. 3. *QuoteLineItem* component to which protocol state machine whose states are defined as sub-states in Fig. 2 is attached. *State2* state machine is attached to a port inter-operating point with *Product* component while Getting Quantity state machine is attached to a port inter-operating point with *Quote* component to describe the external view more precisely by making dynamic constraints in the sequence of operation calls explicit.

Fig. 3 defines each state with invoked operation, pre-condition, and post-condition to be consistent with Fig. 2. Also, from Fig.3, we can catch that the different interfaces of a component interact each other through different ports. In this way, behavior between components which have an assembly composition relationship is described as black-box point view. In Fig. 3, comparing product state is defined as a sub-state since *QuoteLineItem* component has assembly composition relationship with *Product* component. *QuoteLineItem* is a component (i.e. session bean in EJB) which is temporarily connected to *Product* component (i.e. entity bean in EJB).

4 Discussions and Contributions

Though UML 2.0 is still insufficient to describe four levels of contracts proposed by [5], contracts can be more visually described by using UML 2.0 than by using extended forms of association class proposed by [9] extending UML with a new semantic primitive contracts.

It is not necessary for testers and users to know all the detail levels (white-box view) of information. The proper level of information that is required for the component usage and test is the things such as components' primary states, properties, and operations that are required to interact with components and some constraints for related operations. We believe that the protocol state machine is appropriate to provide this level of information. The core problems are to know properties for component composition, states and operations and then to describe them in terms of the protocol state machine and finally to connect it to the external structure diagram of component.

Another problem is to describe the interaction states which are possible between two components. We described the interaction states by using sub-state. Through this technique, we can describe the component structure diagram with protocol state diagram more structured and simple way. The contribution of this paper can be summarized as follows:

- Component composition description in black-box view by using UML 2.0
- Capability to provide the reasonable abstraction of information with users and testers
- Utilizing the black-box view as the basis to generate test cases for testing interoperability between components in the component-based software.

5 Conclusions

We described so far how we could specify component composition focusing on black-box view. Reusing software component cannot succeed if the component does not export clearly stated service guarantees. Indeed, there are many misunderstanding because of informally stated documents. Because a document which is provided by a component provider is too syntactic, it is difficult for clients to get information about what they have to do to use the component. Also, in the case of using formal language, a specification of a component is too difficult for client to understand.

In this paper we tried to specify component composition in a black-box point of view using UML 2.0. Owing to the upgraded UML, we have a chance to specify component contracts in a convenient way. However, the UML 2.0 superstructure specification is somewhat ambiguous when we describe protocol state machine between two connected components. To solve this problem we consider a state connected with other component's provided interface as a sub-state. That sub-state is a protocol state machine connected with the provided interface. After this method we specified connect port, keeping conformation between two ports.

References

1. Selic, B.: What's New in UML 2.0? IBM Rational Software (2005)
2. Jifeng, H., Liu, Z., Xiaoshan, L.: Contract-Oriented Component Software Development. Technical Report 276, UNU-IIST, POBox 3058, Macau, April (2003)
3. UML Modeling Language: Superstructure, Ver 2.0, Aug. (2005)

4. Allen, R.., Garlan, D.: A Formal Basis for Architectural Connection. ACM Transactions on Software Engineering and Methodology, Vol. 6, No. 3, July (1997) 213-249
5. Beubnard, A., Jezequel, J., Plouzeau, N., Watkins, D.: Making Components Contracts Aware. Computer, July, (1999)
6. Jubin, H., Friedrichs, J., the Jalapeno Team: Enterprise JavaBeans by Example. Prentice Hall PTR, (1999) 141-191
7. J. Ivers, Clements, P., Garlan, D., Nord, R., Schmerl, B., Silva, J.: Documenting Component and Connector Views with UML 2.0. Technical Report CMU/SEI-2004-TR-008 ESC-TR-2004-008, (2004)
8. Mayer, B.: Applying Design by Contract. IEEE Computer, Vol. 25, No. 10, Oct. (1992) 40-51
9. Andrade, L. F. and Fiadeiro, J. L.: Interconnecting Objects via Contracts. Lecture Notes in Computer Science, 1723. (1999) 566-583

XML Security Model for Secure Information Exchange in E-Commerce

Kwang Moon Cho

Dept. of Electronic Commerce, Mokpo National University,
61, Dorim, Cheonggye, Muan, Jeonnam, 534-729, Republic of Korea
ckmoon@mokpo.ac.kr

Abstract. The most important technology in the electronic commerce based on Internet is to guarantee the security of trading information exchange. Many technologies are proposed as a standard to support this security problem. One of them is an XML (eXtensible Markup Language). This is used in various applications as the document standard for electronic commerce system. The XML security has become very important topic.

In this paper an XML security model for web services based electronic commerce system to guarantee the secure exchange of trading information is proposed. To accomplish the security of XML, the differences of XML signature, XML encryption and XML key management scheme respect to the conventional system should be provided. The new architecture is proposed based on unique characteristics of XML. Especially the method to integrate the process management system need to the electronic commerce is proposed.

Keywords: Electronic Commerce, XML, Web Sevices, XML Security Model.

1 Introduction

Much of information is propagated by Internet. Internet that is an open communication system provides browsers based on easy protocols and various tools for information handling. Therefore E-Commerce is proliferated. This E-Commerce is based on the standards for document processing in Internet.

The enterprises perform not only the internal activities but also the interactive businesses with other companies to secure the competitive power of them. In general, the trading business between enterprises is performed typically according to the pre-defined business process by exchanging the contracted documents. The purpose of this paper is to propose a business model for B2B environment. This model is based on the business process management system which manages the conventional internal processes of enterprises. This model also analyzes the key elements needed to E-Commerce for inter-enterprises. Especially, the documents and data exchanged between companies is formalized by using the XML messages that are approved as the standard tools for information exchanges. The business processes exchange the XML messages. During all processes, therefore, the efficient business integration

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1003 – 1011, 2006.

may be possible. This model ensures the secure information exchanges which is an essential factor in E-Commerce.

The most threatened factor to the E-Commerce is the security problems. The messages exchanged by an XML message via Internet is not secure because the user authentication is not guaranteed as shown in Figure 1[11]. The E-Commerce should be based on the public key encryption system to authenticate the valid users. The method to ensure the reliability and security of user's public keys is required.

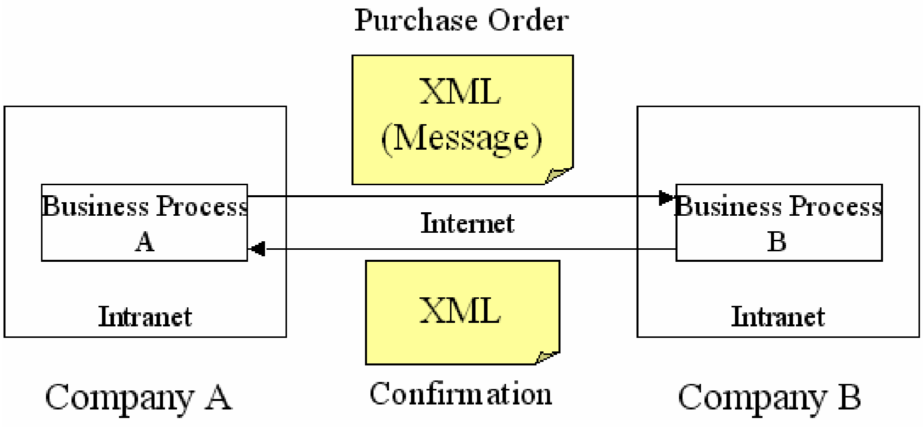


Fig. 1. Unsecured message exchange

Public key infrastructure (PKI) provides secure and reliable method to open the user's public keys to the public [1]. Public key infrastructure has very important roles in Internet E-Commerce. It opens the user's public keys to public in secure and reliable manner.

Since the XML technology is used as the format of message exchange in Internet e-Business, the security for XML documents becoming essential and XML digital signature should be supported for secure E-Commerce [2-6].

In this paper, the security application of E-Commerce is designed which is reliable by using X.509 certificate based on PKI. A web service is designed to implement the PKI-based security application for mutual authentication. The digital signature protocol based on PKI and XML is also designed to solve the security and repudiation problem of message exchange in B2B.

2 Related Technologies

2.1 Public Key Infrastructure

Public key encryption system is an asymmetric system which is based on mathematical functions. It has the pair of keys one is opened to public and the other is saved securely instead of private key encryption system. Then the key is opened is called public key, the other is called private key. The majority security systems for E-Commerce based on public key algorithm because the key management and distribution are difficult. It also resolves the anonymous and user authentication problems.

Public key infrastructure should be constructed based on public key certificates. The certification authority (CA) authenticates the trading subjects. The certification authority creates digital signature by using their own private key and attaches them to the certificate for proving the subject users are valid. The certificate includes the public key of certificate's users and information of subject users.

2.2 Web Services

Web service is a software interface which can be found and called by another programs on the web regardless of location and platforms. Web service is independent on platforms, devices and location. Web service provides dynamic functionality. Web service can be also applied to the conventional systems by low cost.

The web service in E-Commerce is a standardized software technology which combines conventional computer system programs between businesses on Internet. This standard technology enables all business functionalities and services. The web services by using Internet overcome the differences of communications among the heterogeneous operation systems and programming languages. So to speak, the web services are software components which conform e-Business standard and have business logics of Internet.

2.3 XML

XML standard describes the classes of data objects for XML documents. It also describe the operations of computer programs which process these XML documents. XML is an application of SGML (Standard Generalized Markup Language).

XML documents consist of entities which are storage units. The entity contains parsed data or un-parsed data. The parsed data consists of characters. Some of these characters are character data, the others are markups. The markups encode the arrangement plan of physical storage and the description of logical structure. XML provides a mechanism which enforces the arrangement plan of storage and logical structure.

The software module as it called XML processor reads XML document and accesses the content and structure of that.

XML is a standard for organizing the data, XSL (eXtensible Stylesheet Language) is a standard for method to output this data. XSL is a translation technology. XSL is a language to translate each field of XML to relevant tags of HTML and represent to web browser.

XML schema is the term for file to define the structure and content of XML documents. DTD (Document Type Definition) is also a kind of schema, but it has some defects. DTD should be described by E-BNF and so difficult. On the other hand, XML schema can be desctibed just using XML itself. Moreover, XML schema can use various data types that are not supported in DTD. In XML schema the elements can be reused. So to speak, XML schema extended model of DTD. XML schema can define precisely the types of XML documents and the relationships of elements.

XML documents should be parsed to make a tree structure from XML elements. DOM (Document Object Model) is a model to store parsed data as a tree structure and permits accessing particular element. According to DOM, XML documents are analyzed to hierarchical tree structure.

2.4 XML Digital Signature

Recently, XML is in the spotlight as a technology applicable to various applications like B2B and B2C. The importance of security is increased in E-Commerce because the most businesses are processed in electronically. Especially, the standards for security in documents exchanging using XML in E-Commerce having been established. The XML-Signature Group of IETF and W3C recommended the specification for "XML-Signature Syntax and Processing". This specification describes the syntax and processes for XML digital signature.

The following should be considered for security of XML digital signature.

- Confidentiality
- Integrity
- Authentication
- Authorization
- Non-Repudiation

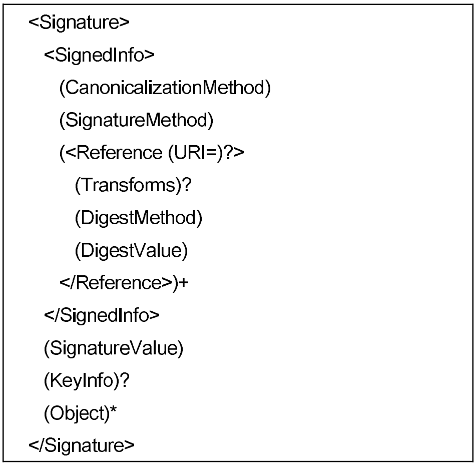
3 XML Security Model

3.1 XML Signature

The syntax of XML signature is a complicated standard to provide various functionalities. It can be applied any signatures because it is designed to have high-level extensibility and flexibility. W3C recommendation defined XML signature syntax and processing rules for them.

Figure 2 shows the XML syntax for digital signature.

XML signature starts with an element <Signature>. The element <Signature> is an important one that consists of signature and identifying the signatures. The element <SignedInfo> lists "the signed information" which are the objects to sign by us. The particular data streams for Digest is represented by the element <References>. The URI (Uniform Resource Identifier) syntax is used to prescribe these streams. The element <KeyInfo> may be used efficiently in automation of XML signature processing because it provides identifying mechanism for verification keys. The element <Object> is a container which can retain any types of data objects. Two elements for <SignatureProperties> and <Manifest> are defined that should be contained in the element <Object>. The element <SignatureProperties> is a pre-defined container to verify signatures. It retains the assertions for signatures. These

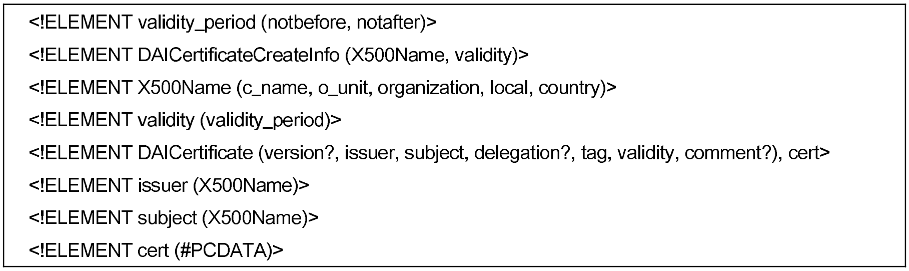


```
<Signature>
  <SignedInfo>
    (CanonicalizationMethod)
    (SignatureMethod)
    (<Reference (URI=)?>
      (Transforms)?
      (DigestMethod)
      (DigestValue)
    </Reference>)+
  </SignedInfo>
  (SignatureValue)
  (KeyInfo)?
  (Object)*
</Signature>
```

Fig. 2. XML syntax for digital signature

assertions may be used to verify the signatures and integrity. The element <Manifest> is used to verify references for application domains. It also provide a convenient method for multiple-signers to sign multiple documents. If the element <Manifest> does not used, the results of signature increase in volume and the performance may be depreciated.

The creation information for certificates and the issued certificates are exchanged in the form of XML documents. The important information is encrypted as a unit of XML element. The DTD for the creation information for certificates is shown in Figure 3.



```
<!ELEMENT validity_period (notbefore, notafter)>
<!ELEMENT DAICertificateCreateInfo (X500Name, validity)>
<!ELEMENT X500Name (c_name, o_unit, organization, local, country)>
<!ELEMENT validity (validity_period)>
<!ELEMENT DAICertificate (version?, issuer, subject, delegation?, tag, validity, comment?), cert>
<!ELEMENT issuer (X500Name)>
<!ELEMENT subject (X500Name)>
<!ELEMENT cert (#PCDATA)>
```

Fig. 3. DTD for XML certificate

3.2 Structure for XML Security

In this paper, the security system is designed based on the web service platform. This system executes and verifies XML signatures independent from the conventional applications. Consider the Purchase Order is submitted by Company A via Internet and is confirmed by Company B as shown in Figure 1. Company A executes digital signature before transmission and Company B confirms after reception. So, the secure

SOAP message exchanges are possible. In this process the Proxy has a role to check the digital signatures under surveillance of delivered messages. The real object to execute and to confirm the digital signature is implemented as a web service. This structure is shown in Figure 4.

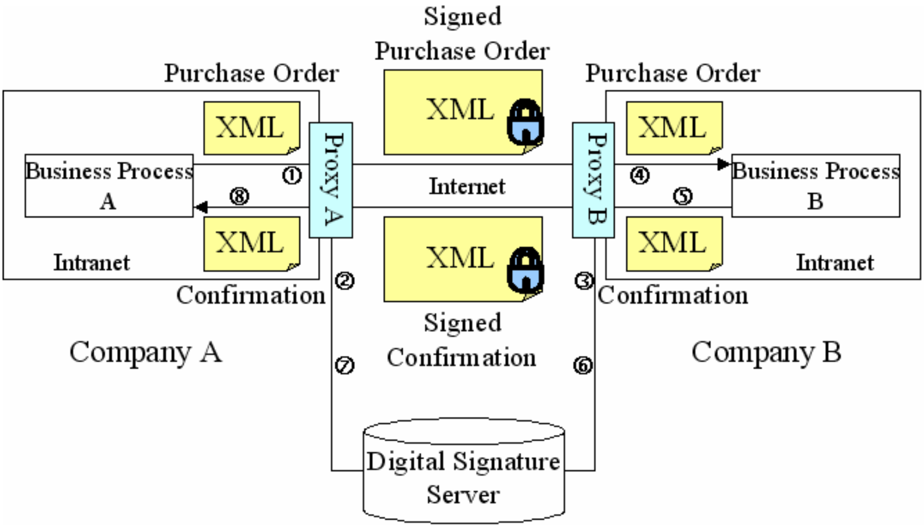


Fig. 4. Secure exchange of XML messages

The following is the procedures for Figure 4.

Step 1. The business process A of company A transmits the message for Purchase Order to business process B of company B.

Step 2. When the purchase is passing proxy A, the digital signature is executed by sending the message to digital signature server.

Step 3. The proxy B of company B receives the signed message and sends it to the confirmation server. The confirmation server verifies the signed message.

Step 4. The verification results are sent to proxy B. If the signature is valid, proxy B removes the signature and sends it to business process B. The information of signer may be preserved.

Step 5. The business process B transacts the message for Purchase Order. The business process B makes a reply message and transmits it to company A.

Step 6. When the reply message is passing proxy B, the digital signature is executed using the private key of company B by sending the message to digital signature server.

Step 7. The company A sends the message from Proxy A to the confirmation server.

Step 8. If the digital signature is valid, the signature is removed from the message and the message is sent to the business process A.

The proxy determines whether it executes digital signature or not by checking the XML messages on network. Consequently, the workflow A and B do not concern the execution and confirmation of signatures. It is a forte that the conventional applications may not be changed.

The content verifier of the proxy server determines whether it needs a digital signature or not by checking the existence of an element <Signature> in XML schema. If it needs, two modules are required. One is to translate the XML message to the form of SOAP message, the other is reverse.

3.3 Execution of Digital Signature

Figure 5 shows an example of the message for Purchase Order with digital signature.

```
<?xml version="1.0" encoding="UTF-8"?>
<Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
<SignedInfo Id="foobar">
<CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
<SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#dsa-sha1" />
<Reference URI="http://www.acompany.com/news/2000/03_27_00.htm">
<DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1" />
<DigestValue>j6lwx3rvEPO0vKtMup4NbeVu8nk=</DigestValue> </Reference>
<Reference URI="http://www.w3.org/TR/2000/WD-xmldsig-core-20000228/signature-sample.xml">
<DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
<DigestValue>UrXLDLBIta6skoV5/A8Q38GEw44=</DigestValue> </Reference>
</SignedInfo>
<SignatureValue>MC0E~LE=</SignatureValue>
<KeyInfo>
<X509Data>
<X509SubjectName>CN=Ed Simon, O=XMLSecurity Inc., ST=OTTAWA, C=CA</X509SubjectName>
<X509Certificate> MIID5jCCA0+gA...lVN </X509Certificate>
</X509Data>
</KeyInfo>
</Signature>
```

Fig. 5. XML Digital Signature

The procedure to execute the message in Figure 5 by digital signature web service is as follows.

Step 1. Determine the object for digital signature. This is given as the form of URI in general.
Step 2. Calculate the value of Digest for each object for signature. The object for signature is defined in the element <Reference> and each Digest is stored in the element <DigestValue>. The element <DigestMethod> defines the algorithm.
Step 3. The element <SignedInfo> contains the elements <Reference> of each objects for signature. The element <CanonicalizationMethod> designates the algorithm that normalizes the element <SignedInfo>.
Step 4. The Digest of the elements <SignedInfo> is calculated and signed, then stored in the element <SignatureValue>.
Step 5. If the information of public key is required, it is stored in the element <KeyInfo>. This is a certificate of X.509 for sender and needed to confirm the digital signature. The procedure for confirmation is shown in Figure 6.

Step 6. Finally, the XML digital signature is generated by including all created elements to the element <Signature>.

Figure 6 shows the procedures to confirm the reliability of digital signature.

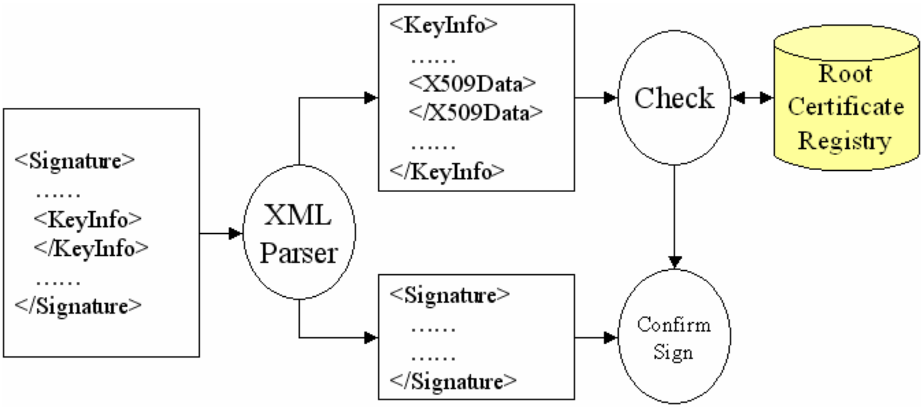


Fig. 6. Confirmation of Reliability for XML Digital Signature

The information of certificates is extracted from the element <KeyInfo> to confirm the generated digital signature. It is compared to the certificate stored in the root certificate registry. Then the reliability is ensured.

4 Conclusion

In this paper, PKI-based digital signature is designed based on XML and web services. It ensures the secure trading and non-repudiation in E-Commerce. The XML digital signature is designed and the operation structure is also proposed when two companies exchange the trading information as the form of XML messages. By using the concepts of proxy and web service, the conventional application programs can be operated without change. All information for document exchange is represented in XML. Only the secret information of XML document is encrypted. Because the digital signature is executed whole document, the security of trading and non-repudiation are guaranteed.

In the future the researches for connecting to the CA, distribution of CRL(Certificate Revocation List) and key renewal for CA are required.

References

[1] RFC: 2560 X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP, Jun. 1996.

[2] www.w3.org, Extensible Markup Language (XML), http://www.w3c.org/XML, Feb. 1998.

[3] www.w3.org, "XML Signature Requirements WD," W3C Working Draft, Oct. 1999.

[4] www.w3c.org, "XML-Signature Syntax and Processing," W3C Recommendation, Feb. 2002.

[5] www.w3c.org, "XML Encryption Syntax and Processing," W3C Working Draft, Oct. 2001.

[6] www.w3c.org, "Decryption Transform for XML Signature," W3C Working Draft, Oct. 2001.

[7] T. Takase et al, "XML Digital Signature System Independent Existing Applications," Proceedings of the 2002 Symposium on Application and the Internet, pp.150-157, 2002.

[8] E. Xavier, "XML based Security for E-Commerce Applications," Eighth Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems, pp.10-17, 2001.

[9] KwangMoon Cho, "Framework of Content Distribution in Mobile Network Environment," Proceedings of the 2003 International Conference on Internet Computing (IC '03), pp.429-434, Jun. 2003.

[10] KwangMoon Cho, "Packaging Strategies of Multimedia Content in DRM," Proceedings of the 2003 International Conference on Internet Computing (IC '03), pp.243-248, Jun. 2003.

[11] KwangMoon Cho, “Web Services based XML Security Model for Secure Information Exchange in Electronic Commerce,” The Journal of Korean Association of Computer Education, Vol.7, No.5, pp.93-99, Sep. 2004.

Design and Implementation of B2B*i* Collaboration Workflow Tool Based on J2EE

Chang-Mog Lee

Division of Electronics and Information Engineering,
Chonbuk National University, 664-14 1ga Duckjin-Dong Duckjin-Gu Jeonju,
Jeonbuk, South Korea
cmlee@chonbuk.ac.kr

Abstract. In this paper, the business process was easily modeled by distinguishing between the business process and work logic. Based on this model, B2B*i* collaboration Workflow modeling tool, which facilitates collaboration, was designed and implemented. The collaboration workflow modeling tool consists of 3 components; business process modeling tool, execution engine and monitoring tool. First, a business process modeling tool is used to build a process map that reflects the business logic of an application in a quick and accurate manner. Second, an execution engine provides a real-time execution environment for business process instance. Third, a monitoring tool provides a real-time monitoring function for the business process that is in operation at the time. In addition to this, it supports flexibility and expandability based on XML and J2EE for the linkage with the legacy system that was used previously, and suggests a solution for a new corporate strategy and operation.

1 Introduction

A new business model is required to reflect changes in the business environment. As the development of information technology accelerates the integration between companies or in a company, the need for a new process has increased. SCM (Supply Chain Management) [1] etc. is the representative example, and its focus is to optimize the logistics of all companies related to the production of a product. The BPM (Business Process Management) is a system that visualizes the business process in and out of a company, executes/controls who is related to business execution and the system according to the process, and controls and optimizes the whole business process efficiently. Activities of a company have become increasingly complicated, and the maximization of domestic and foreign management efficiency has been raised as a key point of success in business. However, the reality is that companies are divided rather than integrated with one another, although customer, partner, managers, information system and business process etc. should be efficiently operated/managed for efficient management. Poor management results in efficiency drains, such as repetition of work, increase of working hours, and data repetition etc. In particular, it is the main cause of poor confrontation with the changes in business

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1012 – 1021, 2006.

process due to a rapidly changing market situation, customers' demand, and changes in inside organization etc. To increase the productivity and efficiency of work, it is necessary to connect and automate these task units focusing on the business process, and BPM (Business Process Management) is the solution that stimulates it.

In the early 1970s, SQL appeared, dramatically simplifying the expansion and correction of a system by dividing data classes and business logic classes. Also, the web innovation in the 1990s simplified the change of work logic by dividing work logic and various expressions of users through huge client/server network technology. Based on this technology, the business application of today came to require process modeling and management that can promptly cope with the business environment and integration between several applications that are operated at various platforms. Based on this, a company can reduce costs by including a legacy system into new system construction and can create new opportunities through B2B collaboration.

In this paper, a J2EE-based B2B*i* collaboration workflow modeling tool, which enables companies to utilize a successful business process management solution to cope with the rapid changing business environment, and which is required to employ a new business strategy and the operation by connecting to the legacy system, was designed and implemented by dividing business process and business logic.

2 Related Works

The workflow management system is the most effective tool to monitor the entire operation of business process and integration of several different information resources [2]. At first, it began as a business process automation tool to support BPR (Business Process Reengineering). However, it has developed into a core technology for the collaboration of distributed computing of groupware, ERP (Enterprise Resource Planning), EAI (Enterprise Application Integration), BPMS(Business Process Management System), and ebXML, e-commerce etc. As workflow has been applied to a lot of fields, several standards were established and new standards will be established in the future. The workflow system is implemented in a centralized manner in order to simplify monitoring. In WfMC(Workflow Management Coalition), a WfMC reference model [3] was constructed as the initial workflow standard. For this model, the standard for workflow engine, circumferential constituents, interface between those and words was established. As the web grew, SWAP (Simple Workflow Access Protocol) [4] was presented as HTTP-based workflow standard. Afterwards, it was integrated to Wf-XML that is XML standard [5]. Wf-XML is XML-based protocol for inter operation between workflow systems, and it provides the minimum message aggregation for inter operation. Next, an agent-based workflow model such as INCAs (INformation CArriers) [6] was suggested. However, although it had an extremely high expandability as a complete distribution workflow model, there was no consideration for monitoring and management function, which were the most important functions in workflow. WONDER system [7] presents the actual implementation of INCAs model, and it had an additional monitor function that the INCAs model does not have. To support workflow on the web, WPDL (Workflow Process Description Language) was established at WfMC, and it was reestablished to XPDL (XML Process Description Language) by applying XML later. XPDL was the definition standard of workflow, but there are unilateral process definition standards such as BPSS (Business Process Specification Schema) of ebXML and PIP (Partner Interface Process) of RosettaNet etc. [8].

The appearance of web service and rapid standardization of the proposed workflow definition standard for various web services in several groups and companies has necessitated a standard to provide workflow function as well as standardization of basic function. The representative examples are IBM WSFL of IBM, and XLANG of Microsoft etc. and these were integrated to BPEL4WS(Business Process Execution Language for Web Services) in recent days [8].

3 Design of B2B*i* Collaboration Workflow Tool

This paper is focused on the design and implementation of a B2B*i* collaboration workflow tool to design, to execute and manage a business process. To do so, a modeler (OrchestraXAStudio) that serves as an automation tool for modeling the business process is required. Any process that is prepared by a modeler should be defined by using XPDL as a workflow process definition language. Therefore, XPDL schema instance generator is required. Also, perfect J2EE-based process execution engine (OrchestraXAServer) for rapid allocation and smooth execution of workflow or process integration application is required too. Finally, a monitoring tool (OrchestraX-AMonitor) that provides a console to inspect and operate the process implemented in this execution engine is required. The Fig. presents the structure of B2B*i* collaboration workflow.

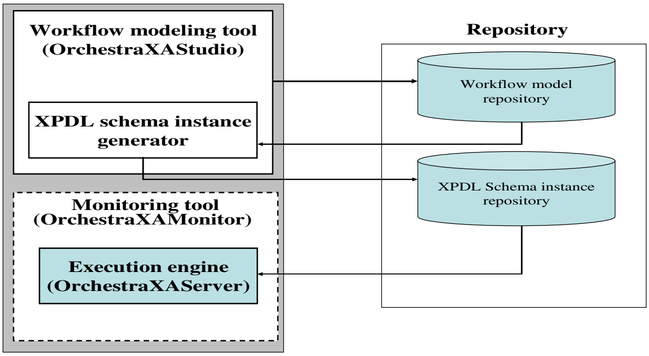


Fig. 1. Architecture of collaboration workflow tool

3.1 Business Process Meta Model

Business process meta model defines the basic aggregation of object and property for process definition exchange. To define a process, objects below should be defined first.

- Workflow process activity
- Transition information

- Details of workflow participant
- Workflow application declaration
- Workflow-related data

In the Fig.2, the workflow process definition consists of more than 1 activity. One activity represents the work that is executed with the verification of the resource that is detailed by the allocation of participants and a computer application that is detailed by the allocation of application program. The workflow participant declaration provides a technology of resources that can serve as a performer of various activities in the process definition

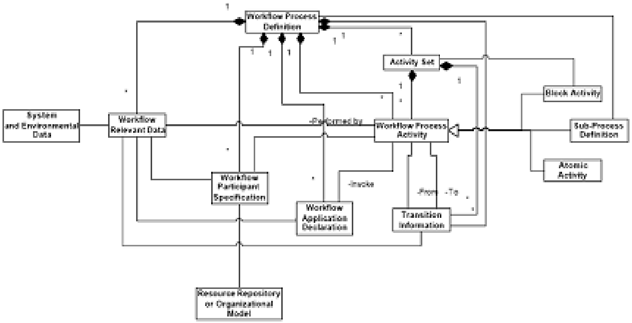

1
Workflow Process Definition
1
1
1
1
1
*
Activity Set
1
1
*
*
Block Activity
*
System and Environmental Data
Workflow Relevant Data
-Performed by
Workflow Process Activity
Sub-Process Definition
*
Atomic Activity
Workflow Participant Specification
*
-Invoke
-From -To
*
*
Workflow Application Declaration
Transition Information
Resource Repository or Organizational Model

Fig. 2. Workflow process definition meta model

3.2 Package Meta Model

Package meta model defines the object and property to exchange or save a process model. It defines various succession rules that correlate workflow-related data that can be defined at a package level rather than at the level of object definition for participant details, application declaration and an individual process definition. The package definition allows multiple process definition property details that can be applied to all the individual process definitions that are included in the package. The package meta model includes the object types listed below.

- Workflow process definition
- Workflow participant details
- Workflow application declaration
- Workflow-related data

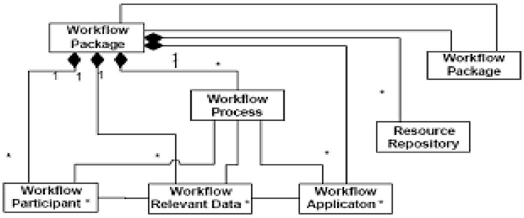


Fig. 3. Package definition meta model

4 Implementation of B2B*i* Collaboration Workflow Tool

4.1 Business Process Modeling Tool

A business process modeling tool is used to design and execute a modeling strategy, policy and process of a company. It enables the field work designer and manager to add or change new work logic easily based on it by suggesting the common activity of companies as a template that is known as an advanced management technique (Best Practice). To complement the steep learning curve that constitutes the major drawback of work process preparation through existing XML-based process sentence preparation, a visual and intuitional modeling interface was selected. By using the modeling tool, works related to process design, the task unit design and process operation can be handled.

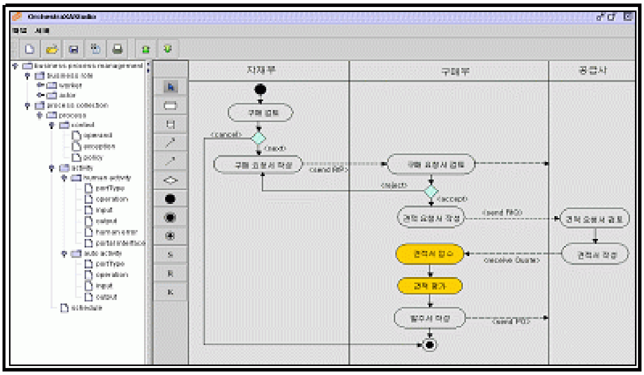


Fig. 4. Main page of business process modeling tool

First, OrchestraXAStudio provides a process modeling tool that even a beginner can easily design complicated business process by using various editing functions such as Drag-and-Drop, Validation and Wizard function etc. It can have an effect of building a process-based system that can actually operate by reflecting work rules. Also, the designed process contributes to a rise in productivity because it can be reused for similar types of business. Second, it helps provide a clear definition of properties such as process participant, execution application, terms of task unit completion, and various branch conditions etc. on task unit in the process. Third, process operation is done according to a rule that is defined in advance when a business process begins. Process participants handle the task through work portal. They are notified of the task in real time, and perform it through applications such as electronic form that is defined in advance.

4.2 Execution Engine (OrchestraXAServer)

The execution engine is a complete J2EE-based process engine for rapid allocation and smooth execution of workflow or process integration application. In general terms, it is called a workflow engine, and it provides a real-time execution environment for business process instance. It is a multi platform and J2EE-based process engine that is neutral in regard to the application server and the main development range exists as shown below.

- Process definition analysis
- Process instance control (generation, activation, pause, close etc.)
- Sequential or parallel operation, deadline scheduling
- Maintenance of workflow control data and workflow-related data
- Interface that calls outside application

4.2.1 Implementation of OrchestraXAServer

OrchestraXAServer can handle all known workflows and process modeling pattern as well as complicated business or application process logic temporarily.

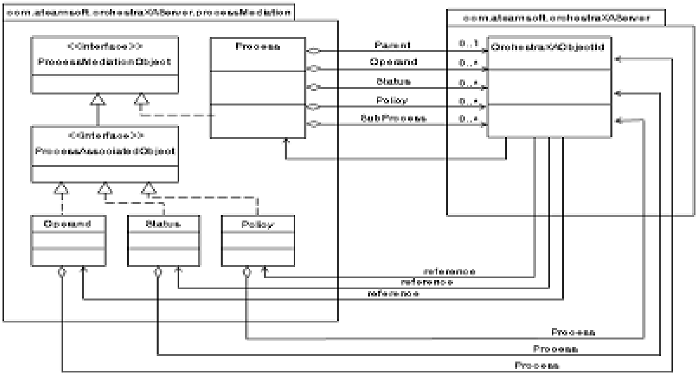


Fig. 5. Object model conceptual scheme

In OrchestraXAServer, business process is modeled into different object types of process object, operand object, status object and policy object.

First, process object is the main constituent of a server object model that includes information about the current status of the process, terms of start, close and change of process, and the relationship that the process object has with other objects. The process object is composed of the relationship and properties stated below.

- Property: None
- Relation

parent process	OrchestraXAObjectId about subordinate process object of the process object
subprocesses	OrchestraXAObjectIds aggregation of subordinate process object of the process object
operands	OrchestraXAObjectIds aggregation of operand objects that are related to process object
statuses	OrchestraXAObjectIds aggregation of status objects that are related to process object
current statuses	Status that is applied to the process at present
policies	OrchestraXAObjectIds aggregation of policy objects that are related to process object

Second, operand object encapsulates temporary data related to the business process that is modeled according to the process object. It can save any value that can be expressed in character string. Also, it can save objects or complicated data structure through XML expression transition or Java serialization. It has property and relations shown below.

- Property

ID	Character string that solely confirms the object in OrchestraXAServer system
label	Short and readable character string that can be used instead of an id to confirm a process. All objects related to the process should have sole label.
description	Text that describes process. It is used by OrchestraXAServer client.
ACL(set of ACEs)	It sets the relation between OrchestraXAServer object and system user.
visible in subtree (true, false)	Truth value that defines the range of operand object
value	Data that is saved in operand object

- Relation

process	Id of process that operand object is related

Third, a status object is a very simple object that is used to trigger the condition of a process object as well as the information how the process is going on and to provide a cause for the execution of policy object. Status objects can be used for various purposes. The status object is closely related to the properties highlighted below.

- Property

ID	Character string that solely confirms the object in OrchestraXAServer system
label	Short and readable character string that can be used instead of an id to confirm a process. All objects related to the process should have a sole label.
description	Text that describes the process. It is used by OrchestraXAServer client.
ACL(set of ACEs)	It sets the relation between OrchestraXAServer object and system user.
visible in subtree (true, false)	Truth value that defines the range of operand object
value	Data saved in operand object

- Relation

process	id of the process that operand object is related

Fourth, a policy object encapsulates the temporary logic related to the process that is modeled. Through this object, a process is customized, and smoothed extensively. A policy object relates Java class and the event that triggers this execution file. A policy object is related to the properties presented below.

- Property

ID	Character string that solely confirms the object in OrchestraXAServer system
label	Short and readable character string that can be used instead of an id to check the process. All objects relate to the process should have a sole label.
description	Text that describes the process. It is used by OrchestraXAServer client.
ACL(set of ACEs)	It sets the relationship between OrchestraXAServer object and system user.
event source	Event source that OrchestraXAEvent object should have for the execution of policy object
event name	Event name that OrchestraXAEvent object should have for execution of policy object (null value indicates that any type of OrchestraXAEvent object can execute policy object)
event attributes	Event property that OrchestraXAEvent object should have for the execution of policy object

language	Language that is used to policy object
source code type (CODE, CLASSNAME, URL)	Property used to determine if character string sourceCode property includes class name, url or original source for executable code of policy object

- Relation

process	id of process that operand object is related

Each policy object consists of one of the two types; executable Java class and BSF script. The source code type property of policy object indicates the approach to the type of policy object and executable command. Fifth, the OrchestraXAServer supports a policy code that can be prepared in various languages, not being limited to Java, JavaScript, Tcl, and Python.

- Use of pre-compiled Java class
Any Java class that implements java.lang.Runnable interface orcom.ateamsoft. orchestraxa.-PolicyScript interface can be called by policy object. Navigator object is used to approach to OrchestraXAServer from policy object.

- Use of Bean Script Framework
BSF allows other programming language to approach to Navigator object. Navigator class is useful to approach to OrchestraXAServer object from policy object. To obtain a Navigator object from Java policy object, the code below is used.

```
Navigator navigator = (Navigator) bsf.lookupBean("navigator");
```

4.3 Business Process Monitoring Tool

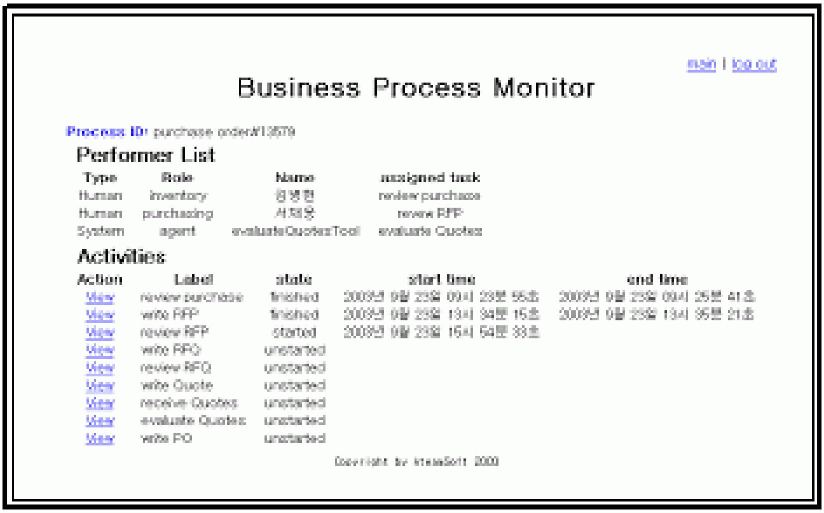
Business process monitoring tool provides a business process operation console. Through this, process the manager traces how both a business unit and individual performs given an activity and task, if the deadline is tight, and it also determines the potential sources that threaten the strategy of the company. A monitoring tool, which is a web-based application, enables process manager to execute management and correction of business item that is in operation anywhere connected by Internet. The measured statistical data provides feedback again to the business process modeling tool for optimized business analysis and improvement of performance.

- Process ID: Process instance recognition string
- Performer List: List of executer related to business process execution

Property	Explanation
Type	It presents if the activity-performing object is a man or automated system. It is indicated as Human if it is a man and as System if it is automated system.
Role	It indicates the role of object that performs the business.
Name	It indicates the name of objects that performs the business.
Assigned task	It indicates the business distributed to the activity.

- Activities: Aggregation of activities that constitute the business process.

Property	Explanation
Action	It outputs detailed contents of the activity
Label	Name of the activity
State	Status of the activity. unstarted. started. finished
Start time	Starting time of activity instance.
End time	Finishing time of activity instance.



main | log out

Business Process Monitor

Process ID: purchase order#13579

Performer List

Type	Role	Name	assigned task
Human	inventory	임병헌	review purchase
Human	purchasing	서재웅	review RFP
System	agent	evaluateQuotesTool	evaluate Quotes

Activities

Action	Label	state	start time	end time
View	review purchase	finished	2003년 9월 23일 09시 23분 55초	2003년 9월 23일 09시 25분 41초
View	write RFP	finished	2003년 9월 23일 13시 34분 15초	2003년 9월 23일 13시 35분 21초
View	review RFP	started	2003년 9월 23일 15시 54분 33초	
View	write RFQ	unstarted		
View	review RFQ	unstarted		
View	write Quote	unstarted		
View	receive Quotes	unstarted		
View	evaluate Quotes	unstarted		
View	write PO	unstarted		

Copyright by InteaSoft 2000

Fig. 6. Individual process level manager interface

5 Tool Comparison and Analysis

Tool Item	ADOME-WFMS	WIDE	Jablonski	OrchestraXA Tool
Modeling method	Suggests a modeling method of execution viewpoint	Cannot suggest a modeling method	Suggests a modeling method of change viewpoint	Suggests a complicated modeling method that supports workflow characteristics
Modeling tool	Supports a process definition language, but lacks modeling tool	Supports a process definition language, but lacks modeling tool	Supports a process definition language, but lacks modeling tool	Implements a tool that can define the relationship between process participant, information system, and business process
Whether or not to apply workflow engine	Not applied	Partially applied to design aspect	Not applied	Implements an engine that supports XPDL
Monitoring tool	Off-line support	Off-line support	Off-line support	A web-based monitoring tool is implemented

6 Conclusion

This paper implemented functions required to implement BPM according to process management cycles from the viewpoint of customer. The results consist of OrchestraXAStudio, which is a business process modeling tool, OrchestraXAServer, which is a business process execution engine, and the component of OrchestraXAMonitor, which is a business process modeling tool. First of all, OrchestraXAStudio, which is business process modeling tool, is a process mapping tool that is the basis of BPM. It supports standardization of mapping schematization through a database and common ownership of the web, rather than a document, for the process of a company. OrchestraXAStudio clearly presents and schematizes the correlation between strategy resource processes and provides process-centered business management and constant change management. OrchesetraXAServer, which is a business process execution engine, is business process automation tool that executes a standardized process, and J2EE-based process engine for rapid allocation and smooth execution of workflow or process integration application. J2EE-based process engine that is a multi platform and neutral to application server provides the core argument of this paper. OrchestraXAServer can handle all known workflows and process modeling patterns, as well as temporarily complicated business or application process logic. OrchestraXAMonitor, which is a business process monitoring tool, can control the cycle-time of the process in operation as well as process and KPI (Key Performance Indicator) etc. through monitoring. Especially, as it can verify the process improvement effect in advance by using cycle-time simulator etc., it can provide a 'What-If' analysis function according to an improvement scenario and can maximize process improvement activity of BPR (Business Process Reengineering) as well as BPM simultaneously.

References

1. George L. Kovács and Paolo Paganelli, "A planning and management infrastructure for large, complex, distributed projects—beyond ERP and SCM", Computers in Industry, Vol. 51, Issue 2, June 2003, pp. 165-183
2. E. Cream, Web Services Essentials, O'Reilly, First Edition, 2002.
3. J. O. Kephart and D. M. Chess, "The Vision of Autonomic Computing", IEEE Computer, January, 2004.
4. Workflow Management Coalition, "The Workflow Reference Model", Workflow Management Coalition, January 1995.
5. G. A. Bolcer, G. Kaiser, "SWAP: Leveraging the Web to Manage Workflow", IEEE Internet Computing, Vol. 3, No. 1, January-February 1999.
6. Workflow Management Coalition, "Workflow Standard-Interoperability Internet Wf-XML Binding", Workflow Management Coalition, July 1998.
7. D. Barbara, S. Mehrota, and M. Ruinkiewicz, "INCAs: Managing Dynamic Workflows in Distributed Environments", Journal of Database Management, 7(1):515, 1996.
8. BEA, IBM, and Microsoft, "BPEL4WS Version1.0", August 2002, available at http://www.106.ibm.com/developerwork/webservices/library/ws-bpel

Traffic-Predicting Routing Algorithm Using Time Series Models

Sangjoon Jung[1], Mary Wu[2], Youngsuk Jung[3], and Chonggun Kim[2]

[1] School of Computer Engineering, Kyungil University,
712-701, 33 Buho-ri, Hayang-up, Gyeongsan-si, Gyeongsang buk-do, Korea
sjjung@kiu.ac.kr
http://www.kiu.ac.kr
[2] Dept. of Computer Engineering, Yeungnam University,
712-749, 214-1, Dae-dong, Gyeongsan-si, Gyeongsangbuk-do, Korea
mrwu@yumail.ac.kr, cgkim@yu.ac.kr
http://nety.yu.ac.kr
[3] School of Computer Engineering, Kyungwoon University,
730-850, 55 Indeok-ri, Sandong-myeon, Gumi-si, Gyeungsang buk-do, Korea
ysjung@ikw.ac.kr
http://ce.ikw.ac.kr

Abstract. A routing algorithm is proposed that analyzes network traffic conditions using time series prediction models and determines the best-effort routing path. To predict network traffic, time series models are developed under the stationary assumption, which is evaluated using the Auto Correlation Function (ACF) and Partial Auto Correlation Function (PACF). Traffic congestion is assumed when the predicted result is larger than the permitted bandwidth. Although the proposed routing algorithm requires additional processing time to predict the number of packets, the packet transmission time is reduced by 5~10% and the amount of packet loss is also reduced by about 3% in comparison to the OSPF routing algorithm. With the proposed routing algorithm, the predicted network traffic allows the routing path to be modified to avoid traffic congestion. Consequently, the traffic predicting and load balancing by modifying the paths avoids path congestion and increases the network performance.

1 Introduction

Monitoring network traffic involves inspecting the amount of data and types of packet that are being transferred[1,2], while network monitoring provides collected data and an analysis of such data[3], including where congestion has occurred at certain nodes, which allows a network traffic manager to control the routing path when congestion occurs[4,5].

As such, the proposed routing algorithm attempts to analyze network conditions using time series models and then determine the best-effort routing decisions. To predict traffic based on time series models, the model must satisfy the stationary assumption, obtained using the ACF and PACF[6], as if the time series does not satisfy the

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1022 – 1031, 2006.

stationary assumption, the Mean square error (MSE) between the real values and the predicted values will be large, indicating that the predicted results are inaccurate. If traffic congestion can then be assumed when the predicted result is larger than the permitted bandwidth, traffic predictions can be used to modify the routing path to avoid traffic congestion. As a result, predicting network traffic can facilitate more effective management of the routing tables and the ability to make best-effort decisions to increase the network performance.

2 Time Series Models and Routing Protocols

2.1 Time Series Models

A time series is a sequence of observations that are ordered in time (or space). If observations of a particular phenomenon are made at intermittent time intervals, it makes sense to display the data in the order in which it occurred, especially since successive observations are also likely to be inter-dependent[5]. Thus, time series data analyzed according to a specific function can be made into the following time series models[6].

2.1.1 AR(Auto Regressive) Model

The auto regressive approach is based on the premise that each observation in a time series is related in a consistent and identifiable way to one or more previous observations in the same series. The form of this model is as follows.

$$Z_t' = \phi_1 Z'_{t-1} + \phi_2 Z'_{t-2} + \cdots + \phi_p Z'_{t-p} + a_t,$$

$$\text{here, the white noise is } a_t = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{z_t^2}{2\sigma^2}}. \quad (1)$$

2.1.2 MA(Moving Average) Model

The moving average is a form of average that has been adjusted to allow for seasonal or cyclical components in a time series. As such, the function of this model is to smooth the original time series by averaging a rolling subset of elements from the original series, consisting of an arbitrary selection of consecutive observations. Thus, the moving average process can be thought of as the output from a linear filter with a transfer θ(B), when white noise is inputted.

$$Z'_t = a_t - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \cdots - \theta_q a_{t-q}. \quad (2)$$

2.1.3 ARMA(Auto Regressive Moving Average) Model

To obtain an accurate model, the inclusion of both AR and MA terms is sometimes necessary. The form of this model is as follows.

$$\begin{aligned} Z_t' = {} & \phi_1 Z'_{t-1} + \phi_2 Z'_{t-2} + \cdots + \phi_p Z'_{t-p} + a_t \\ & - \theta_1 a_{t-1} - \theta_2 a_{t-2} - \cdots - \theta_q a_{t-q}, \end{aligned} \quad (3)$$

$$\phi_1, \theta_1, a_t : \text{Estimated values}$$

An autoregressive model of order *p* is conventionally classified as AR(*p*), while a moving average model with *q* terms is classified as MA(*q*). Thus, a combination model containing *p* autoregressive terms and *q* moving average terms is classified as ARMA(*p*,*q*).

2.1.4 ARIMA(Auto Regressive Integrated Moving Average) Model

The process of ARIMA modeling allows the two modeling approaches to be integrated. If the object series is differenced *d* times to achieve stationarity, the model is classified as ARIMA(*p*,*d*,*q*), where the symbol "I" signifies "integrated." In theory, ARIMA models are the most general class of models for forecasting a time series that can be stationarized by such transformations as differencing and logging[5]. The form of this model is as follows.

$$\begin{aligned} Z_t &= \phi_1 Z_{t-1} + \phi_2 Z_{t-2} + \cdots + \phi_p Z_{t-p} + u_t \\ &\quad - \theta_1 u_{t-1} - \theta_2 u_{t-2} - \cdots - \theta_q u_{t-q} , \\ \phi(L) z_t &= (L) u_t \quad \textit{even if}, \quad \phi(L) = 1 - \phi_1 L - \cdots - \phi_p L^p , \\ &\qquad\qquad\qquad\qquad\quad \theta(L) = 1 - \theta_1 L - \cdots - \theta_q L^q , \\ \textit{Therefore}&,\ \phi(L) \nabla^d y_t = (L) u_t \cdot \end{aligned} \tag{4}$$

2.2 ACF(Autocorrelation Function) and PACF(Partial Autocorrelation Function)

The stationary assumption is regarded as the previous step in the prediction process in order to make an accurate prediction. Stationary time series data have a constant mean, constant variance, and the covariance is independent of time. An assumption of stationary data is essential for predicting network traffic[5,6]. The stationary assumption can be evaluated using the ACF and PACF[5,6], which should both satisfy the stationary assumption. Autocorrelation refers to the correlation of the time series with its own past and future values. As such, the ACF(Autocorrelation Function) is a tool used to assess the degree of dependence included in the time series data. Thus, when trying to fit a model to time series data, the function is applied to the data. Meanwhile, the PACF(Partial Autocorrelation Function) refers to the correlation between observations Z_t and Z_{t-k} after removing the linear relationship between the observations from Z_{t-1} to Z_{t-k+1} .

2.3 Routing Algorithms

Existing routing protocols can be divided into two categories, DV(Distance Vector) algorithms and LS(Link State) algorithms, depending on how the routing information is updated to manage the shortest distance to the destination[7,8]. A DV algorithm determines the optimized route using the next hop count and shortest distance, then periodically exchanges this routing information with its neighbors[7,8]. In contrast, an LS algorithm manages the overall network structure information and partial cost per link and broadcasts this router state information to others. A shortest path first algorithm is then used to find the best route to the destination by classifying the cost information. The representative link state protocol is OSPF(Open Shortest Path First)

[9,10]. Although DV algorithms are used as the primary routing mechanism in most current networks, they still have a number of weaknesses. For example, route problems, such as local broadcasting, can occur due to the use of local information. In DV routing, a change in the routing topology due to a node becoming overloaded and timing out can cause neighboring nodes to change the path used to route packets to particular destinations. Since this causes a new broadcast to each subsequent neighbor, the control information being broadcast can quickly overload the network if the topology keeps changing[11].

3 Analysis of Network Traffic Sample

In the present study, network traffic was analyzed using time series models that satisfied the stationary assumption to ensure an appropriate model for predicting the total number of packets.

3.1 Gathering Time Series Data and Model Building

The traffic monitoring system was connected to an intra-network to gather network packets. The system then revealed details of the network traffic based on analyzing the data gathered from the source, destinations, and total number of packets transferred at each node. The traffic monitoring was undertaken for 1 year from July 2003 to August 2004. The number of packets was a trace of the real traffic collected from the same network. To predict the amount of packets, the AR (equation 1), MA (equation 2), ARMA (equation 3), and ARIMA (equation 4) were all applied, then the most suitable model in terms of expressing the nature of the traffic was identified for forecasting the traffic. The satisfaction of the stationary assumption was also confirmed using the ACF and PACF, which were calculated using SPSS 11.0 software[12], as the calculation process is very complex and the SPSS results are trustworthy.

3.2 Proof of Stationary Assumption and Prediction of Traffic

To determine the adaptability of the time series models, the stationary assumption was examined. Here, SPSS was used to identify the proper model for predicting the network traffic. Unfortunately, the complete data collected for the year did not satisfy the stationary assumption. Thus, for better results, the data was sub-divided on a monthly, weekly and daily basis. However, only the daily basis revealed significant ACF and PACF results, as shown in the following table.

Table 1. Analysis of time series models based on verification of stationary assumption using data classified on daily basis

Model	AR	MA	ARMA	ARIMA	Inappropriate
Results	275	10	26	15	11

Most of the data classified on a daily basis satisfied the stationary assumption, and applying AR(1), the AR model when the lag number was 1, was identified as the best model for predicting the total number of packets in the near future.

4 Traffic-Predicting Routing Algorithm Using Time Series Models

When satisfying the stationary assumption, a time series model can be used to predict the network traffic, and an accurate prediction value can ensure an efficient path is chosen to avoid congestion. Thus, if the predicted value is larger than the permitted bandwidth, the router modifies the routing table to avoid congestion. The following then describes the prediction-based routing update operations in the routers. First, the routers gather routing information and update the routing table. When links states are changed and packets transmitted, the routers predict the future packet amount and update the routing table if the predicted value is larger than the permitted bandwidth. Here, the update algorithm is executed every 30 seconds after packets are transferred, and the AR model applied when the lag number is 1.

4.1 Proposed Traffic-Predicting Routing Algorithm

The proposed routing algorithm makes use of the AR(1) model to forecast the total number of packets in the near future, as shown in Fig. 1.

```
program obtain_prediction_value
global real array $(X)_{24:0}$
global real parameter $n \leftarrow 0$
real parameter $X_{t+1}$, $\phi_1$, $a_t$
real $X_{sum} \leftarrow 0$, $X_{mean}$
integer $i$
$X_n \leftarrow$ obtain the time series data
for $i$=0 to $n$ do
    $X_{sum} \leftarrow X_{sum} + X_i$
end do
$X_{mean} \leftarrow X_{sum} / (n+1)$
for $i$=1 to $n$ do
    $\phi_1 \leftarrow (X_i - X_{mean}) * (X_{i-1} - X_{mean}) / (X_{i-1} - X_{mean})^2$
end do
$a_t \leftarrow 1/\mathrm{sqrt}(2\pi) * \exp(-X_n^2/2)$
$X_{t+1} \leftarrow \phi_1 * X_n + a_t$
$n \leftarrow n+1$
end program obtain_prediction_value
```

Fig. 1. Proposed traffic-predicting routing algorithm

First, each node gathers packets when packets are transferred, and the parameter is estimated using the previously calculated data mean. Next, the routing algorithm applies the AR(1) model to forecast the total number of packets using equation (1). The following step then determines the appropriate model, where parameter ϕ_1 is calculated using a Least-square estimation, as it has already been established that sample autocorrelations and partial autocorrelations calculated from the residuals of a Least-square fit are asymptotically equivalent to those obtained from actual data.

$$\hat{\phi}_1 = \frac{\sum_{t=2}^{n}(X_t - \overline{X})(X_{t-1} - \overline{X})}{\sum_{t=2}^{n}(X_{t-1} - \overline{X})^2}.$$

In the next step, when fitting a regression model to time series data, the possibility of autocorrelation in the error term should always be considered, and a reasonable approach to identify an appropriate model for the error a_t is to obtain the Normal distribution.

$$a_t = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{X_t^2}{\sigma^2}}, \quad X_t \sim N(0,1) \cdot$$

Finally, the algorithm obtains the number of packets X_{t+1} in the near future by multiplying the parameter with the current X_t and adding the error term a_t. Forecasting using AR models with time series errors is straightforward when the future values X_{t+1} of the explanatory variables are assumed to be deterministic functions (1).

$$X_{t+1} = \hat{\phi}_1 X_t + a_t \cdot$$

The O-notation indicates the significance of a particular measure in a program, i.e. the prediction operation requires $O(n)$ time, and since n is never more than 24, the program is not complicated. Thus, when a router receives packets and needs to determine their paths, the routing algorithm can update rapidly.

4.2 Cost Decisions Using Prediction

OSPF is the most widely used link-state routing protocol, which determines the routing cost based on calculating the bandwidth. Thus, to include the predicted traffic in this calculation, if the packet prediction value is over 90% of the bandwidth usage, the cost of the routing is increased by 10% (k_1=0.9, k_2=0.1). The following shows the cost calculation algorithm.

```
Step1. $Cost_{ospf} = 10^8 / Bandwidth_{medum}$
Step2. Prediction Bandwidth($PB$) = $X_{t+1}$* 8/1,000,000
Step3. **if** Prediction Bandwith($PB$) > $Bandwidth_{medium} * k_1$
           **then** $Cost_{pr} = Cost_{ospf} + Cost_{ospf} * k_2$
```

4.3 Routing Update Algorithm

Routing decisions are based on a routing table that in this case is created using the Dijkstra algorithm. Packets are only transferred when every node has a routing table. After the packets are transferred, the proposed algorithm is then applied every 30 seconds. If the predicted result is larger than the permitted bandwidth, the routing cost is increased by 10%, and the increased cost transferred to every node to find the best route. The routing table is then changed. The routing algorithm used to update the routing table is as follows.

```
Step 1. Calculate the routing cost using the Dijkstra algorithm.
Step 2. Make routing tables in every router.
Step 3. while (Generate packets && Timer == 30 seconds) {
              1. Execute the prediction algorithm.
              2. Generate the prediction result($X_{t+1}$).
              3. Execute the cost calculation algorithm.
              4. Send the $Cost_{pr}$ to every router.
              5. if (Link state is changed == true)
           break;
          }
Step 4. Transmit the changed cost.
```

5 Experiments and Analysis

An NS-2 simulator was used to analyze the effect of the proposed traffic prediction. Every node had a routing table based on the cost of the bandwidth. The OSPF routing protocol (LS) was then compared with the prediction-based algorithm (PR).

5.1 Experimental Environments and Evaluations

The topologies used in the experiments were as follows.

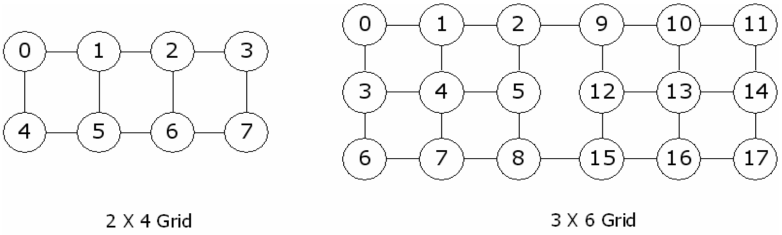


Fig. 2. Simulation topologies

To evaluate the proposed algorithm, the LS algorithm and proposed routing algorithm were applied to two types of topology. However, despite the simple computing procedure, NS-2 proved to be exceedingly slow and had difficulty handling even a moderate sized topology with the desired the simulation time. Implementing topologies larger than 2x4 or 3x6 proved to be nearly impossible. Therefore, small topologies were simulated, including real-world traffic generators, to evaluate the routing performance and reveal potential weaknesses in the proposed algorithm. Plus, since these topologies cannot simulate a routing environment, a mesh-type topology was also designed As such, the performance was evaluated in a 2x4 and 3x6 grid, while the mesh-type topology was a 2x2 grid and 3x3 grid connected by 2 links, as shown in Fig. 2. The 2x4 gird represented the internal routing paths, while the 3x6 grid represented the external routing paths. The following table presents the experimental environment.

Table 2. Simulation environments

<table>
<tr><th colspan="2"></th><th>2 x 4 Grid</th><th>3 x 6 Grid</th></tr>
<tr><td colspan="2">Number of nodes</td><td>8</td><td>18</td></tr>
<tr><td colspan="2">Bandwidth and delay time</td><td colspan="2">1.5 Mbps 10ms
10 Mbps 10ms</td></tr>
<tr><td colspan="2">Executing time</td><td colspan="2">400 seconds</td></tr>
<tr><td colspan="2">Traffic types</td><td colspan="2">FTP, TELNET, CBR, Exponential, Pareto</td></tr>
<tr><td colspan="2">Maximum nodes</td><td colspan="2">4</td></tr>
<tr><td rowspan="2">Maximum packet sizes per links</td><td>1.5 Mbps</td><td colspan="2">189,440 Bytes</td></tr>
<tr><td>10 Mbps</td><td colspan="2">1,249,280 Bytes</td></tr>
</table>

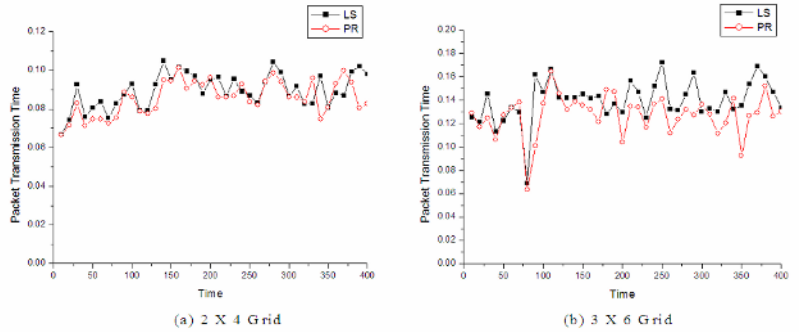
The OSPF routing algorithm and proposed routing algorithm were both applied to evaluate which was better.

5.2 Experimental Results

Various factors were compared to evaluate the routing performance of the two algorithms.

5.2.1 Packet Transmission Time

An important goal of packet transmission is how fast a packet arrives at the destination node from the transmitting node. Therefore, the packet transmission time was compared with a 1.5 Mbps bandwidth in the each topology, as depicted in Fig. 3.

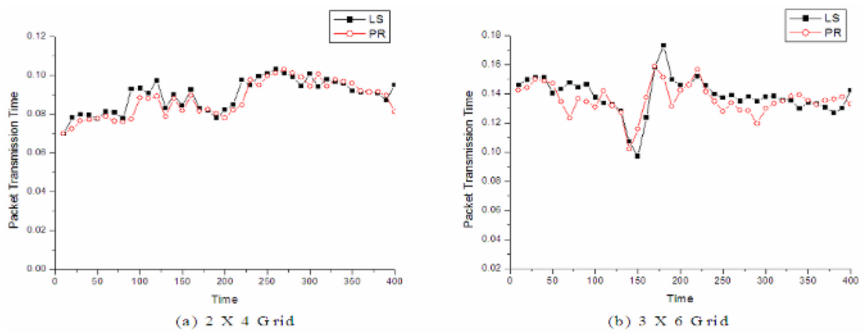


(a) 2 X 4 Grid (b) 3 X 6 Grid

Fig. 3. Comparison of packet transmission time with 1.5Mbps in each topology

The trend for the PR algorithm was similar to that for the LS algorithm over time. Fig. 3 shows that the PR algorithm was slightly lower than the LS algorithm, and as time passed, the PR routing algorithm ensured that the packets arrived rapidly at their destination.

Fig. 4 also shows that the PR algorithm was better than the LS algorithm. Thus, it was concluded that the packets with the PR routing algorithm arrived rapidly at their destination. The transmission time for the PR algorithm was also better than that for the LS algorithm, as seen by the average packet transmission time for the two routing algorithms shown in table 3.



(a) 2 X 4 Grid

(b) 3 X 6 Grid

Fig. 4. Comparison of packet transmission time with 10Mbps in each topology

Table 3. Average packet transmission time

Bandwidth	Topology	LS	PR
1.5 Mbps	2 x 4 Grid	0.089373	0.085724
	3 x 6 Grid	0.139585	0.127583
10 Mbps	2 x 4 Grid	0.089645	0.087484
	3 x 6 Grid	0.138749	0.135924

In table 3, the PR algorithm reduced the packet transmission time, as it reduced the transmission latency caused by hard traffic.

5.2.2 Amount of Packet Loss

After applying the routing algorithms to each topology, the amount of packet loss was compared between the two algorithms. The packet transmission results for the two algorithms in each topology are shown in table 5.

Table 4. Amount of packet loss

Band-width	Compared Item	2x4 Grid		3x6 Grid	
		LS	PR	LS	PR
1.5 Mbps	Sum of packets	532,862	541,792	813,925	819,248
	Sum of drop packets	13,245	6,432	24,286	5,981
	Drop rate	2.486%	1.187%	2.984%	0.730%
10 Mbps	Sum of packets	3,553,143	3,621,875	5,424,354	5,462,648
	Sum of drop packets	84,248	40,880	98,609	73,891
	Drop rate	2.371%	1.129%	1.818%	1.353%

Using the prediction method when the packets were transferred reduced the amount of packet loss, and the PR algorithm had lower packet loss rates than the LS algorithm in each topology. Therefore, the proposed algorithm produced a higher successful packet transmission rate than the LS algorithm.

6 Conclusion

While the OSPF routing algorithm provides an equal path multi path(ECMP), it does not change the route until the routing path is altered. As a result, routing traffic congestion occurs when hard traffic happens without switching the route. Accordingly, this paper proposed a new routing algorithm that attempts to analyze network conditions using time series models. As a result, traffic predictions allow the routing path to be modified to avoid traffic congestion. The proposed routing algorithm is characterized by an increasing packet amount rather than a constant packet amount. Simulation results show that routing paths guarantee the rapid transmission of packets during the provision of packet services. Traffic congestion is also avoided, thereby reducing the volume of packet loss. As the routing information packets are broadcast periodically every 30 seconds, this does increase the network traffic. However, when compared with other routing algorithms, the amount of routing packets is similar to that with the RIP routing algorithm. Thus, traffic prediction using time series models is helpful in improving the packet service. Future studies will attempt to reduce the routing information packets in the proposed algorithm.

References

1. Ryan Kastner, Elaheh Bozorgzadeh and Majid Sarrafzadeh : Predictable routing, Computer Aided Design, IEEE/ACM International Conference, (2000) 5-9
2. Xun Su and Gustavo de Veciana : Predictive routing to enhance QoS for stream-based flows sharing excess bandwidth, Computer Networks, Volume 42, Issue 1, (2003) 65-80
3. W. Leland, et al.: On the Self-Similar Nature of Ethernet Traffic (extended version), IEEE/ACM Transactions of Networking, Vol. 2, no. 1, (1994) 1-15
4. Wilinger,W., Wilson,D., Taqqu, M.: Self-similar Traffic Modeling for Highspeed Networks, ConneXions, (1994)
5. Yantai Shu, Zhigang Jin, Lianfang Zhang, Lei Wang : Traffic Prediction Using FARIMA Models, IEEE International Conference on Communications, (1999) 891-895
6. William W. S. Wei : Time Series Analysis, Addison-Wesley, (1990)
7. Hedrick, C.: Routing Information Protocol(RIP), Network Information Center, RFC 1058, (1988)
8. F. C. M. Lau, Guihai Chen, Hao Huang and Li Xie: A distance-vector routing protocol for networks with unidirectional links, Computer Communications, Volume 23, Issue 4, (2000) 418-424
9. J. Moy: OSPF version 2, RFC 1583, (1994)
10. G. Michael Schneider, Tamas Nemeth: A simulation study of the OSPF-OMP routing algorithm, Computer Networks, Volume 39, Issue 4, (2002) 457-468.
11. D. Kim, K. Ryu, and Y. Cho: A new routing control technique using active temporal data management", Journal of Systems and Software, Volume 51, Issue 1, (2000) 37-48
12. SPSS for windows Trends Release 11.0, SPSS Inc. (2001)

A Study on Software Architecture Evaluation

Gu-Beom Jeong[1] and Guk-Boh Kim[2]

[1] Dept. of Computer Engineering, Sangju National University,
Sang Ju, South Korea
jgb97@sangju.ac.kr
[2] Dept. of Computer Engineering Daejin University, Po Chon,
South Korea
kgb@road.daejin.ac.kr

Abstract. This paper presents an approach to evaluate software architecture. Our approach is divided into three main areas of activities : the work involved in preparation, execution, and completion of the evaluation. Through performing these activities, architectural evaluation can be systematically performed centering around architectural design decisions that have profound impacts on the achievement of quality attributes.

1 Introduction

With the popularity of the Web and the Internet, the technology to develop software has advanced rapidly to improve the quality of software, produce it on time, and efficiently adapt to the change in requirements. In this situation, it needs to be noted that the quality attributes of any nontrivial system are principally determined by its architecture[1,2]. Furthermore, it is always more cost-effective to evaluate software quality as early as possible in the life cycle of the system[3]. The obvious risk is that potentially large amounts of resources will have been put into building a system which does not fulfill its quality requirements[4]. For this reason, it is important to evaluate and determine whether a system is destined to satisfy its desired qualities or not before it is built[3,5].

Although methods for evaluating software architecture with respect to software quality attributes exist (e.g. [1],[4],[6],[7]), theses evaluation techniques have too many limitations for wide-spread applications. A typical limitation for some of these techniques is that they require considerable effort from the software engineer for creating specifications and making predictions[4,6,7]. Other techniques often require information about the system under development that is not available during architectural design because of an ambition for accurate results to guarantee the functionality or quality required of a system[4,7]. Finally, more recently developed techniques using scenarios have a number of uncertainties associated with the steps such as scenario elicitation[1,2,3].

To address these problems, this paper introduces our approach to software architecture evaluation. Our approach allows software architects to assess architecture qualitatively. It is more effective than quantitative assessment or theoretical assessment

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1032 – 1041, 2006.

[4,7]. In addition, our approach systematically bridges quality requirements of software systems with their architecture centering around architectural design decisions. To do this is worthy work[2].

2 Related Work

2.1 Software Architecture Model: The 4+1 View Model

The 4+1 view model produces a mechanism to allow us to separate concerns while building or analyzing an architecture. Architects capture their design decisions in four views and use the fifth view to illustrate and validate them. Each view addresses a specific set of concerns as follows[8,9]:

– *Logical View*. The logical view includes a set of abstractions necessary to depict the functional requirements of a system at an abstract level.
– *Process View*. The process view describes the design's concurrency and synchronization aspects.
– *Development View*. The development view describes the software's static organization in its development environment.
– *Physical View*. The physical view describes the mapping of the software onto the hardware and reflects its distributed aspect.
– *Use case View*. The use case view describes an abstraction of important requirements as use case.

2.2 Existing Approaches to Software Architecture Evaluation

Several research communities have developed techniques to perform architectural evaluation. In this section, the characteristics and limitations of some representative approaches are briefly illustrated.

Techniques evaluating a specific quality attribute. Several research groups have developed techniques used for the specification and assessment of their particular quality requirements. Of those techniques, some techniques[10,11] have adopted statistical models, e.g. Markov Chain Model and Queuing models. On the other hand, the ADL(Architecture Description Language) research groups have developed various kinds of languages to represent architectural information relevant to their specific quality attributes, and they have analyzed architecture using them[12]. But these approaches tend to require considerable effort from the software engineer for creating specifications and making predictions.

Techniques using simulations or prototypes. These techniques require that the main components of the architecture are implemented, and other components are simulated resulting in an executable system[4]. But, these techniques require information about the system under development that is not available during the architectural design.

Scenario-based evaluation techniques. A scenario-based technique is used to attempt to reduce the problematic nature of evaluating a high-level design with respect to software quality attributes[4]. To assess a particular quality attribute, a set of scenarios has to be developed to make concrete the actual meaning of the quality requirements. The technique focuses on architectural features that will reveal design biases and flaws early in the life cycle of the system. In these techniques, however, there are a number of uncertainties such as the granularity of representation and how representative the scenarios are in respects to their evaluation steps[13]. In [1,2,3,4], scenario-based techniques have been introduced.

3 Software Architecture Evaluation

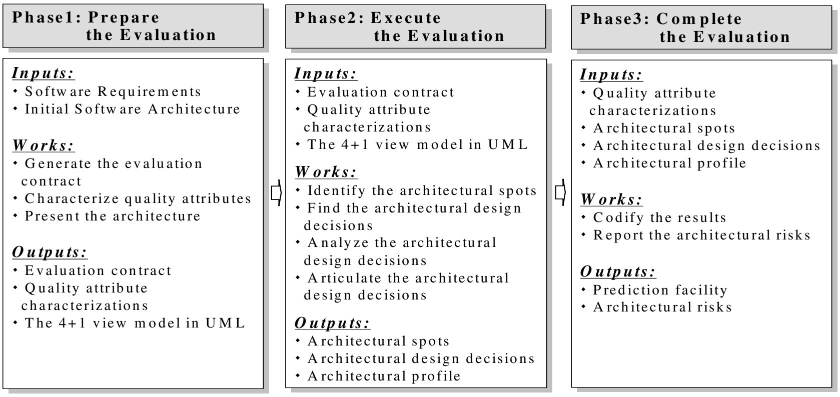
3.1 An Overview of Our Approach

Our approach is divided into three main areas of activities: preparing the evaluation, executing the evaluation, and completing the evaluation. More specifically, the activities are as follows:

Phase1. Prepare the Evaluation. In this phase, we plan the goals and scope of our architectural evaluation and present the architecture as well. Our approach uses the features of a 4+1 view model of architecture with UML, and handles the functional and quality requirements separately. Also, quality attributes are more explicitly characterized by their quality requirements.

Phase2. Execute the Evaluation. During this phase, we find architectural design decisions and analyze them. Also, we articulate them to understand the relationships among them with respect to quality attributes.

Phase3. Complete the Architecture. In this phase, the previous results are incorporated into a structured form to bridge quality attributes with their architecture. Also, the architectural risks, which mean potentially problematic architectural design decisions, are reported. They allow us to predict the quality attributes of software architecture or help us improve it in order to meet its quality requirements. Figure 1 shows the overall activities of the proposed approach.



Phase1: Prepare the Evaluation

Inputs:
- Software Requirements
- Initial Software Architecture

Works:
- Generate the evaluation contract
- Characterize quality attributes
- Present the architecture

Outputs:
- Evaluation contract
- Quality attribute characterizations
- The 4+1 view model in UML

Phase2: Execute the Evaluation

Inputs:
- Evaluation contract
- Quality attribute characterizations
- The 4+1 view model in UML

Works:
- Identify the architectural spots
- Find the architectural design decisions
- Analyze the architectural design decisions
- Articulate the architectural design decisions

Outputs:
- Architectural spots
- Architectural design decisions
- Architectural profile

Phase3: Complete the Evaluation

Inputs:
- Quality attribute characterizations
- Architectural spots
- Architectural design decisions
- Architectural profile

Works:
- Codify the results
- Report the architectural risks

Outputs:
- Prediction facility
- Architectural risks

Fig. 1. Overall process of our approach, which shows the inputs to each phase, the works in each phase, and the outputs from each phase

3.2 Steps of Our Approach

Step1. Generate the evaluation contract. Generating an evaluation contract determines how many requirements will be handled and the kinds of quality attributes that will be dealt with. It is essential to clarify the scope and goals of an evaluation because an architectural evaluation will be derived from them. During this step, expectations from the evaluation can be concluded and negotiated. To produce such a contract as shown in Fig. 2, the following activities need to be performed:

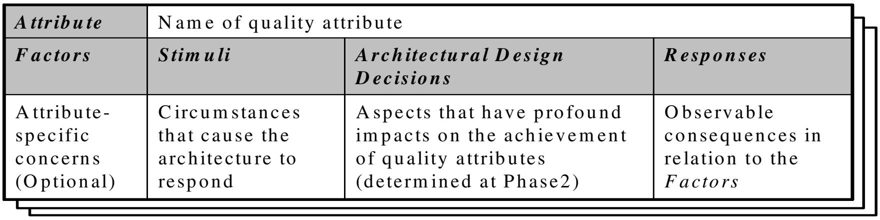
Goals	Names of quality attributes (e.g., Performance, Flexibility, ...)		
Contexts	Performance	QR1, QR2, QR3, ...	
	...	...	
Functional Requirements (FRs)	P1: Description of primary function1 P2: Description of primary function2 ...		1 (# of Rank) 2 ...
	M1**:** Description of mandatory function1 M2: Description of mandatory function2 ...		P1 P2 ...
	O1**:** Description of optional function1 O2: Description of optional function2 ...		P1 P1 ...
Quality Requirements (QRs)	QR1**:** Description of quality requirement1 QR2: Description of quality requirement2 ...		
QRs-FRs Relations	QR1	M1, M2, M4, O1, ...	
	QR2	M1, M3, M4, O2, ...	
	...	...	

Fig. 2. *Evaluation contract*, which defines the scope and goals of architectural evaluation

- *Define the template for requirements.* The template for requirements is a simple form for documenting quality requirements and functional requirements of a system separately. When the template for requirements is defined, each quality requirement is related to one or more functional requirements. This separation of requirements helps identify quality attributes and find architecturally significant parts afterwards.
- *Determine the scope of functional requirements.* The row named *Functional Requirements (FRs)* in Fig. 2 indicates the evaluation scope of functional requirements. To determine this, the primary functions are first ranked according to their relative importance, and its mandatory and optional functions are determined by them. Then the *FRs* row is completed by the subset of the functional requirements.
- *Determine the scope of quality requirements.* The row named *Quality Requirements (QRs)* in Fig. 2 indicates the evaluation scope of quality requirements. It depends on the scope of functional requirements in terms of the relations between quality requirements and functional requirements. Thus, the *QRs* row is completed by the subset of the quality requirements, and the row named *QRs-FRs Relations* can be also completed.

- *Identify the quality attributes*. The row named *Goals* in Fig. 2 indicates the required qualities of an architecture. The quality attributes can be determined by their quality requirements. Also, the row named *Contexts*, indicating architectural concerns associated with each quality attribute, is completed by the quality requirements per each quality attribute.

Step2. Characterize quality attributes. Since architectural evaluation focuses on quality attributes, it is important to have clear and informative characterizations for each quality attribute. To facilitate using the wealth of knowledge that already exists in the various quality attribute research groups and to facilitate eliciting the appropriate attribute-related information, we define and use a template for characterizing quality attributes as shown in Fig. 3. This allows us to organize the important characteristics of a quality attribute. As Fig. 3 shows, quality attribute characterizations include the following contents similar to what have been introduced in [5,10]:



Attribute	Name of quality attribute		
Factors	***Stimuli***	***Architectural Design Decisions***	***Responses***
Attribute-specific concerns (Optional)	Circumstances that cause the architecture to respond	Aspects that have profound impacts on the achievement of quality attributes (determined at Phase2)	Observable consequences in relation to the *Factors*

Fig. 3. *Quality attribute characterizations*, which concretely define the characteristics of quality attributes

- *Attribute*. Attribute denotes the particular quality attribute.
- *Factors*. Factors denote attribute-specific concerns, which are conceived by the *Contexts* contents of the evaluation contract. This is an optional element.
- *Stimuli*. Stimuli are specific circumstances that cause the architecture to respond, which can also be identified from the *Contexts* contents of the evaluation contract.
- *Architectural design decisions*. Architectural design decisions are aspects that have a profound impact on the achievement of quality attributes. They are determined during the Step5 and Step6 of Phase2.
- *Responses*. Responses denote observable consequences in relation to the *Factors*.

Step3. Present the architecture. It is noted that the large variance of a quality assessment based on architectural analysis is associated with the granularity of the system description necessary to perform an evaluation. In this respect, the 4+1 view model of architecture is one solution to obtain good architectural documentation. Through using the 4+1 view model of architecture, quality attributes can be widely covered, and the architecture can be described at the appropriate level of abstraction for wide-spread application. In the 4+1 view model, architectural information is summarized as shown in Table 1. For instance, logical structure and the inter-connection mechanism of a system can be described in logical view, and component concurrency and their synchronization can be described in process view. Also, architectural style or pattern can be described in each view. Due to this consensus of architecture-level information, a more explicit evaluation is possible.

Table 1. Architectural information in the 4+1 view model

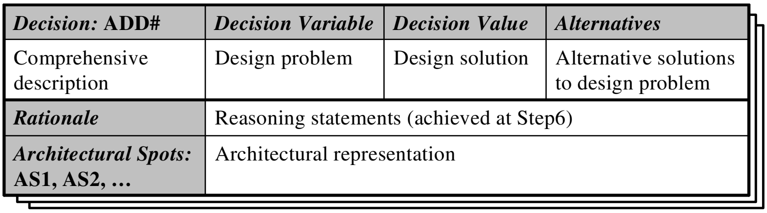
Views	Architectural information
Use case	-an abstraction of important requirements -the principal purposes of system
Logical	-a set of abstractions and their relationships -logical structure of a system -logical inter-connection mechanism
Development	-the organization of the actual software elements -the components' interfaces
Process	-the concurrency and synchronization
Physical	-physical distribution of software components

Step4. Identify the architectural spots. Once the functional requirements have been defined and an architecture has been presented, we can find architectural spots associated with the functional requirements of the evaluation contract. Architectural spots are the significant parts of architectural designs with respect to the evaluation. To find them, we perform the following activities.

- *Identify use cases describing the primary purposes of the system.*
- *Determine views from quality attributes related to the functional requirements.*
- *Identify the architectural spots in relation to the use cases from the views.*

The result from these steps is the architectural spots on which the evaluation is focused.

Step5. Find the architectural design decisions. Since architectural design decisions have a profound impact on the achievement of quality attributes, it is very important to find and analyze them during architectural evaluation. Figure 4 shows how architectural design decisions can be expressed. One architectural design decision expresses its comprehensive description, the *Decision Variable*, the *Decision Value*, the *Alternatives*, the *Rationale*, and the *Architectural Spots*. In particular, Fig. 4 shows that architectural design decisions can be defined as the selection of solutions to design problems faced during architectural design. According to this definition, this step contains two subsequent activities that find the architectural design decisions from explicit architectural spots.



Decision: **ADD#**	***Decision Variable***	***Decision Value***	***Alternatives***
Comprehensive description	Design problem	Design solution	Alternative solutions to design problem
Rationale	Reasoning statements (achieved at Step6)		
Architectural Spots: **AS1, AS2, ...**	Architectural representation		

Fig. 4. *Architectural design decisions*, which are the aspects that have a profound impact on the achievement of quality attributes. Here, ADD# denotes the identifier of Architectural Design Decision and AS# denotes the identifier of Architectural Spot.

- *Determine the decision variables*, which denote design problems. Possible questions from the architectural spots can be reasonably raised based on the architectural information of Table 1 such as "What are the big parts?" and "How are they connected?". The decision variables are then determined by one or more design issues in relation to the *Stimuli* of quality attribute characterizations.
- *Determine the decision values*, which denote the selected solutions to each problem. With the determination of decision values from the architectural spots, their architectural alternatives can also be found by using the knowledge from designs or competing architectures.

Step6. Analyze the architectural design decisions. Once the architectural design decisions have been found, they must be separately analyzed. Analysis in this step does not entail detailed simulation or precise mathematical modeling. It is more of a qualitative analysis for revealing areas of risk in a design. When we reason through the decisions, we may use the various design theories[14,15,16] or refer to other alternatives from competing architectures. We elicit statements about the design decisions' effects on particular quality attribute and the row labeled *Rationale* of Fig. 4 is then completed. As an aftereffect, both the documentations of quality attribute characterizations (see Fig. 3) and architectural design decisions (see Fig. 4) can be completed.

Step7. Articulate the architectural design decisions. Here, the dependencies among the decisions are determined. That is, a decision is identified in how it will affect quality attributes in relation to other decisions. During this step, an architectural profile is generated as shown in Fig. 5. It illustrates how the decisions of the *Subjects* affect the decisions of the *Objects* with respect to particular quality attributes. Therefore, it helps in determining the tradeoffs among quality attributes and understanding the effects made by changing architectural design.

Architectural Design Decisions		*Objects*			
		ADD 1	*ADD* 2	*ADD* 3	...
Subjects	*ADD* 1		(QA1,+)		
	ADD 2	(QA2,-) (QA3,+)			
	ADD 3				(QA1,+)
	...				

Fig. 5. *Architectural profile*, which is a report representing the relationships among architectural design decisions acquired through architectural analysis. *QAi* denotes the particular quality attribute and +/- denotes the positive(+) or negative(-) effect on *QAi*.

Step8. Codify the results. Systematically codifying the relationships between architecture and quality attributes greatly enhances the ability of understanding and controlling quality attributes in the architecture. Thus, we synthetically organize some useful materials via architectural design decisions (the *ADD Layer*) as shown in Fig. 6. To do this, we define three kinds of layers (i.e., the *QA Layer*, the *ADD Layer*, and the *AS Layer*) and their relations in terms of previously discussed works. The end result of this step is not only helpful in identifying areas of risk in a design, but also useful in understanding an architecture's fitness with respect to its quality attributes.

In addition, it can support gradual evaluation and controlling of the complexity by the extension of the evaluation scope.

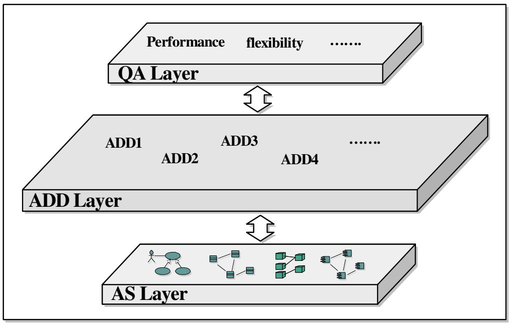


Fig. 6. *Prediction facility*, which is a synthesized structure to bridge quality requirements of software systems with their architecture. Here, *QA*, *ADD*, and *AS* denote Quality Attribute, Architectural Design Decision, and Architectural Spot, respectively.

Step9. Report the architectural risks. Finally, the findings of architectural risks are reported as shown in Fig. 7. As Fig. 5 and Fig. 6 show, both the *architectural profile* and *prediction facility* contribute to revealing design biases or flaws. The architectural risks must contain two major parts: the *condition* and the *consequences*. The *condition* indicates what is currently causing concern, and the *consequences* indicate the design decision's impacts on the relevant quality attribute. This explicit documentation of architectural risks can support understanding of the architectural risks and planning for risk mitigation.

RISK: **RSK#**	Natural description for understanding this risk
Architectural Design Decision	ADD#
Related Attributes	Performance, …
Condition	What is currently causing concern
Consequences	The impacts on quality attributes
Architectural Spots	AS1, AS2, …

Fig. 7. *Architectural risks*, which are potentially problematic architectural design decisions

So far, we have briefly introduced our approach to software architecture evaluation. In summary, architecture evaluation is systematically prepared, executed, and completed centering around architectural design decisions. During these activities, decision-centric software architecture evaluation is qualitatively performed, and architectural risks are finally reported.

3.3 Artifacts of Our Approach

Figure 8 shows the artifacts and their relationships to the evaluation activities. As Fig. 8 shows, the architectural evaluation is initiated by software requirements and initial software architecture. Through performing the phases of architectural evaluation, the useful intermediate materials such as an evaluation contract, quality attribute characterizations, architectural design decisions, and architectural profile, are defined or generated systematically. And lastly the prediction facility and the architectural risks are presented based on these intermediate artifacts.

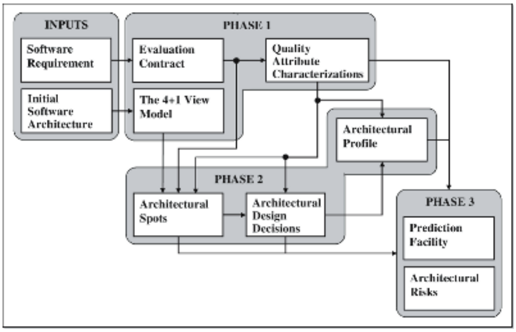


Fig. 8. Core artifacts of architectural evaluation

4 Conclusions and Future Work

We presented an approach to decision-centric architecture evaluation for quality software development. In this approach, we proposed a more systematic guideline process for qualitatively evaluating software architectures that centers around architectural design decisions, which have a profound impact on the achievement of quality attributes. Also, we proposed a more concrete documentation of evaluation artifacts for understanding conceptual flows of our approach and following them more easily. In addition, we proposed using the 4+1 view model in UML to reduce the ambiguities in the architectures studied, for example, when the evaluation is performed and what is evaluated has not been clearly defined. Compared with related works described in Section 2, the features of our approach can be summarized as follows:

Evaluation process is well defined. The inherent ambiguities of architectural evaluation, for example, what levels of detail in software architecture or software requirements are appropriate, can be reduced by two factors: One is determining the set of architectural information in the architecture and the other is defining the information to be achieved from software requirements. For this reason, we proposed using the 4+1 view model in UML during architectural evaluation, and we defined the useful artifacts such as an evaluation contract and quality attribute characterizations. Based on the proposed model and the artifacts, decision-centric architectural evaluation can be more explicitly performed without the difficulties presented by other techniques such as ADL techniques, simulation and prototype techniques, and scenario-based techniques.

Evaluation artifacts are concretely documented. During the process, their artifacts were concretely specified. It helps in understanding conceptual flows of our approach. Also, it is possible to bridge quality requirements of software systems with their architecture and to understand architectural risks through acquiring explicit insights about architecture.

Quality attributes are widely covered. In our approach, various quality attributes to be addressed in the 4+1 view model, e.g., performance, maintainability, availability, security, reliability, and interoperability, can be qualitatively evaluated. The trade-offs among them can also be addressed. In addition, the architecture can be described at an appropriate level of abstraction for wide-spread application through using UML specifications.

In the future, we will show this approach's substantiality through lots of experimental results. Also, we will discuss some interesting issues such as handling ambiguities in software requirements and dealing with bad architectural design decisions.

References

1. Kazman, R., et al., "The Architecture Tradeoff Analysis Method", The 4th IEEE International Conference on Engineering of Complex Computer Systems, August 1998, pp.68-78.
2. Dobrica, L. and Niemela, E., "A Survey on Software Architecture Analysis Methods", *IEEE Transactions on Software Engineering*, IEEE Computer Society, Vol. 28, No. 7, July 2002, pp.638-653.
3. Bass, L., Clements, P., and Kazman, R., *Software Architecture in Practice*, Addison-Wesley, 1998.
4. Bosch, J., *Design and Use of Software Architectures*, Addison-Wesley, 2000.
5. Clements, P., Kazman, R., and Klein, M., *Evaluating Software Architectures*, Addison-Wesley, 2002.
6. Allen, R., "A Formal Approach to Software Architectures", CMU-CS-97-144, Carnegie Mellon University, May 1997.
7. Abowd, G., et al., "Recommended Best Industrial Practice for Software Evaluation", CMU/SEI-96-TR-025, Carnegie Mellon University, January 1997.
8. Kruchten, P., "The 4+1 View Model of Software Architecture", *IEEE Software*, Vol. 12, No. 6, November 1995, pp42-50.
9. Eriksson, H. and Penker, M., *UML Toolkit*, Addison-Wesley, 1998.
10. Klein, M. and Kazman, R., "Attribute-Based Architectural Styles", CMU/SEI-99-TR-022, Carnegie Mellon University, October 1999.
11. Inverardi, P., Mangano, C., Russo, F., and Balsamo, S., "Performance Evaluation of a Software Architecture: A Case Study", *Proceedings of the 9th International Workshop on Software Specification and Design*, April 1998, pp.116-125.
12. Medvidovic, N., "A Classification and Comparison Framework for Software Architecture Description Languages", UCI-ICS-97-02, University of California, Irvine, February 1996.
13. Clements, P., Bass, L., Kazman, R., and Abowd, G., "Predicting Software Quality by Architecture-Level Evaluation", *Proceeding of the 5th International Conference on Software Quality*, October 1995, pp.485-498.
14. Gamma, E., Helm, R., Johnson, R., and Vlissides, J., *Design Patterns*, Addison-Wesley, 1995.
15. Buschmann, F., et al., *Pattern-Oriented Software Architecture – A System of Patterns*, John Wiley & Sons, 2000.
16. Schmidt, D., Stal, M., Rohnert, H., and Buschmann, F., *Pattern-Oriented Software Architecture – Patterns for Concurrent and Networked Objects*, John Wiley & Sons, 2000.

RFID-Based ALE Application Framework Using Context-Based Security Service[*]

Jungkyu Kwon and Mokdong Chung

Dept. of Computer Engineering, Pukyong National University,
599-1 Daeyeon-3Dong, Nam-Gu, Busan, 608-737, Korea
puker@puker.net, mdchung@pknu.ac.kr

Abstract. We propose an RFID-based ALE Application Framework (AAF) providing context-aware security services, which could dynamically adapt security policies. From the proposed framework, we can construct RFID application fast and efficiently through general, reusable, and extensible API due to the software reusability. The proposed model consists of an adaptive security level algorithm based on MAUT and Simple Heuristics. The security level algorithm could adopt diverse security services according to the contextual information in the network environment. Therefore, the proposed model is expected to provide more flexible security management in the heterogeneous network environments.

Keywords: RFID, ALE, Context-awareness, Kerberos, MAUT, Simple Heuristics.

1 Introduction

Researches and developments for ubiquitous computing environment which is human centered computing paradigm are widely spread. One of them is the research and development of RFID (Radio Frequency Identification) technology. The EPC (Electronic Product Code) Network, which was developed by the Auto-ID Center and now managed by EPCglobal, is suggested to enable all objects in the world to be linked via the Internet [1, 4]. The functions of the EPC Network using RFID tags are as follows: recognize, identify all objects, track and trace, monitor, trigger events and actions on those objects, and offer real-time view of assets and inventories throughout the global supply chain.

Current computing environment is composed of the heterogeneous networks, where there are diverse characteristics of the network properties such as transmission media types, bandwidth, device types, and so on. And the properties of these heterogeneous networks are dynamically changing in accordance with the changes of the environment. The existing security models, however, could not provide an appropriate security management since they have only static security management policies.

[*] This work was supported by the Regional Research Centers Program(Research Center for Logistics Information Technology), granted by the Korean Ministry of Education & Human Resources Development.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1042 – 1051, 2006.

Thus, we propose an adaptive security management model for ALE Application Framework (AAF), which could dynamically adapt security policies according to the changes of dynamic network environments. We use MAUT (Multi-Attribute Utility Theory) and Simple Heuristics to introduce contextual information to determine appropriate security policies.

The structure of the paper is as follows. Section 2 discusses related work. Section 3 deals with AAF for providing context-aware security services. Section 4 shows context-aware security services in detail. Section 5 concludes this paper with the future work.

2 Related Work

2.1 EPCglobal Network

EPCglobal network shows a new standard, called ‘Application Level Events (ALE)’ which is developed from the concept of a middleware, ‘Savant’. The role of the ALE is to provide a means to process the event data which are collected by the RFID reader and to deliver them to the higher-level applications [1, 5].

On looking into the structure and components of the EPCglobal, RFID reader delivers identified tag data to the middleware, ALE Engine. Middleware is trying to filter out the various overlapped tag data, and transmits accumulated/filtered tag data index to EPCIS or applications.

ALE, a kind of an interface, defines API (Application Programming Interface) regarding on accumulation, filtering, counting and logging the transferred tag data from the RFID readers.

ALE application defines its own Event Cycle, registers it through API, and receives the EPC index which might be used in an application through subscription/publication API.

2.2 The Context-Aware Computing

In Dey’s definition [2], context may include physical parameters (type of network, physical location, temperature, etc) and human factors (user's preferences, social environment, user's task, etc), and is primarily used to customize a system behavior according to the situation of use and/or users' preferences. Context-aware computing is a mobile computing paradigm in which applications can discover and take advantage of contextual information (such as user location, time of day, nearby people and devices, and user activity) [6]. Thus this paradigm may provide the user with the suitable service which could be appropriate to the user by combining contextual information and the user input.

2.3 Multi-Attribute Utility Theory and Simple Heuristics

MAUT (Multi-Attribute Utility Theory) [7,12] is a systematic method that identifies and analyzes multiple variables in order to provide a common basis for arriving at a decision. As a decision making tool to predict security levels depending on the security context, MAUT suggests how a decision maker should think systematically about identifying and structuring objectives, about vexing value tradeoffs, and about balancing various risks.

The Center for Adaptive Behavior and Cognition is an interdisciplinary research group founded in 1995 to study the psychology of bounded rationality and how good decisions can be made in an uncertain world. This group studies Simple Heuristics [8]. One of them is Take-The-Best which tries cues in order, searching for a cue that discriminates between the two objects. It serves as the basis for an inference, and all other cues are ignored.

2.4 The Contribution of This Paper

This paper has two contributing factors. The first one is as follows. In this paper, we will show the RFID-based ALE Application Framework (AAF) which relies on the EPCglobal Network and the ALE specification, and could construct an RFID application in the diverse domains with ease due to the software reusability.

The second contribution is as follows. The existing security management model usually considers static security policies. Thus, it will be quite difficult to deal with the changes of the environment promptly and appropriately. For instance, SSL/TLS (Secure Socket Layer/Transport Layer Security) [9,11] selects the most secure Cipher Suite in the Cipher Suite List transmitted from the client. But this static policy could result in an excessive overload and long latency to the users who have lower level computing devices. However, if we adopted the adaptive security management policies, we could reduce the latency time to those users.

3 The ALE Application Framework

3.1 The Architecture of ALE Application Framework (AAF)

AAF is architecture of a framework which provides API (Application Programming Interface) with the user, and allows the user to develop the ALE application with ease. The global architecture of the ALE Application Framework is shown in Figure 1. EPC Event is delivered from the Data Manager to the Event Manager where the Logical Event is constructed and it is delivered from the Event Manager to the Business Process Manager.

3.1.1 The Business Process Layer

The Business Process Layer implements a business facility of the application. This layer contains the Business Rule which implements the process with business logic, and contains the Business Context which gives the meaning in a business process. The application programmers can reuse the Domain Free Rule Base without knowing the details of the domain. Only the Domain Sensitive Rule Base, however, might be changed according to the specific domain.

The Logical Event which is generated from the Event Layer expresses meaningful expression according to the Business Rule. We can deliver it to EPCIS or different legacy system through the Data Layer. The representative Business Rule definitions are BPML 1.0 and the process specification of ebXML [3,13].

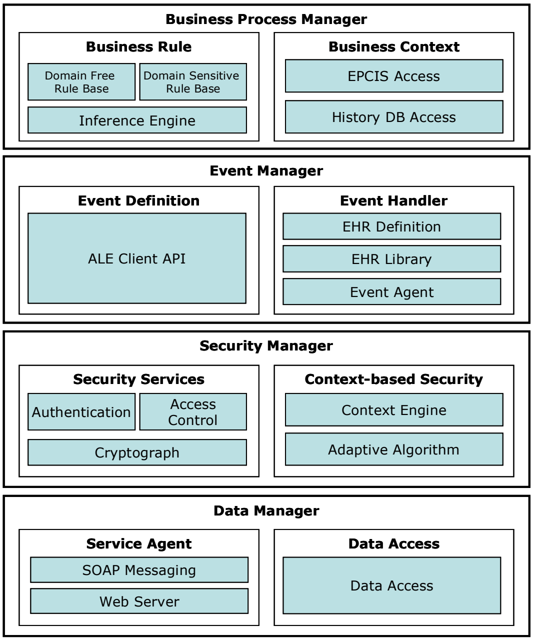


Fig. 1. ALE Application Framework

3.1.2 The Event Layer

The Event Layer implements the definition facility to deal with EPC Event. This layer contains event definition which defines EPC Event, and the Event Handler which deals with EPC Event coming from the ALE Manager which is a middleware located below this framework. The Event Handler produces logical events by using several ECReports coming from the ALE Manager. The Logical Events are extensible by adding appropriate events.

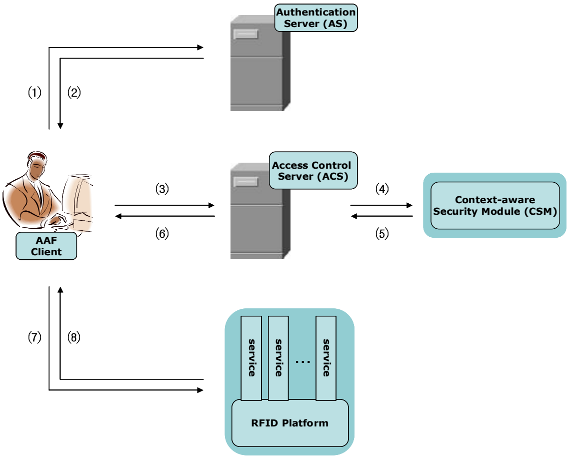

Authentication Server (AS)
(1)
(2)
AAF Client
(3)
(6)
Access Control Server (ACS)
(4)
(5)
Context-aware Security Module (CSM)
(7)
(8)
service
service
...
service
RFID Platform

Fig. 2. Integrated Authentication Model for RFID Services

3.1.3 The Data Layer

The Data Layer offers the facility to access to an outside system. This layer is data access component which takes charge of input and output processing of data accesses to the database, and includes Web Services component to be flexible enough to an outside system.

We send ECSpec to the ALE manager to define the request from the user, and get ECReport from the ALE manager. Thus, this system is responsible for the communication among legacy systems or EPCIS and other systems.

Table 1. Authentication protocol based on Kerberos and Context-awareness

<table>
<tr><td colspan="2">Authentication Service Exchange</td></tr>
<tr><td colspan="2">(1) C → AS: $ID_c // ID_{acs} // TS_1$
(2) AS → C: $\mathrm{Ek_c}[K_{c,acs} // ID_{acs} // TS_2 // L_2 // Ticket_{acs}]$
$\mathrm{Ticket_{acs}} = \mathrm{Ek_{acs}}[K_{c,acs} // ID_c // AD_c // ID_{acs} // TS_2 // Lifetime_2]$</td></tr>
<tr><td colspan="2">Context-aware Ticket Granting Service Exchange</td></tr>
<tr><td colspan="2">(3) C → ACS: $ID_v // Ticket_{acs} // Authenticator_c$
(4) ACS → CSM<i>: Contextual Information (In the same server)</i>
(5) CSM → ACS: <i>Adaptive Security (In the same server)</i>
(6) ACS → C: $\mathrm{Ek_{c,acs}}[K_{c,v} // ID_v // TS_4 // Ticket_v]$
$\mathrm{Ticket_{acs}} = \mathrm{Ek_{acs}}[K_{c,acs} // ID_c // AD_c // ID_{acs} // TS_2 // Lifetime_2]$
$\mathrm{Ticket_v} = \mathrm{Ek_v}[K_{c,v} // ID_c // AD_c // ID_v // TS_4 // Lifetime_4]$
$\mathrm{Authenticator_c} = \mathrm{Ek_{c,acs}}[ID_c // AD_c // TS_3]$</td></tr>
<tr><td colspan="2">Client/Server Authentication Exchange</td></tr>
<tr><td colspan="2">(7) C → V: $Ticket_v // Authenticator_c$
(8) V → C: $\mathrm{Ek_{c,v}}[TS_5 + 1]$
$\mathrm{Ticket_v} = \mathrm{Ek_v}[K_{c,v} // ID_c // AD_c // ID_v // TS_4 // Lifetime_4]$
$\mathrm{Authenticator_c} = \mathrm{Ek_{c,v}}[ID_c // AD_c // TS_5]$</td></tr>
<tr><td colspan="2">Notations</td></tr>
<tr><td>$\mathrm{ID_c}$, $\mathrm{ID_{acs}}$, $\mathrm{ID_v}$</td><td>Identifier of Client, ACS, and Server</td></tr>
<tr><td>$\mathrm{AD_c}$</td><td>Network address of C</td></tr>
<tr><td>$\mathrm{TS_k}$</td><td>Timestamp</td></tr>
<tr><td>$\mathrm{Lifetime_k}$</td><td>Lifetime</td></tr>
<tr><td>$\mathrm{K_{a,\,b}}$</td><td>Session key between <i>a</i> and <i>b</i></td></tr>
<tr><td>$\mathrm{Ticket_{acs}}$</td><td>Authentication granting ticket</td></tr>
<tr><td>$\mathrm{Ticket_v}$</td><td>Service granting ticket</td></tr>
<tr><td>Authenticator</td><td>Authenticating information</td></tr>
<tr><td>AS</td><td>Authentication Server</td></tr>
<tr><td>ACS</td><td>Access Control Server</td></tr>
<tr><td>CSM</td><td>Context-aware Security Module</td></tr>
</table>

3.1.4 The Security Layer

Figure 2 shows an integrated authentication model for the RFID services.

This model modified the basic functions of Kerberos [10] and adopted the context awareness of MAUT and Simple Heuristics. Table 1 shows the detailed protocol for the following scenario which is corresponding to that of Figure 3. The context awareness of MAUT and Simple Heuristics is described in section 4 in detail.

(1) Client requests authenticating-granting ticket.
(2) AS returns authenticating-granting ticket.
(3) Client requests service-granting ticket.
(4) ACS requests service-granting ticket.
(5) CSM returns adaptive service-granting ticket.
(6) ACS returns adaptive service-granting ticket.
(7) Client requests service.
(8) Optional authentication of server to client.

3.2 The Characteristics of ALE Application Framework (AAF)

The characteristic of AAF is that we can construct RFID application fast and efficiently with the low cost through general, reusable, and extensible API. AAF utilizes Web Services, thus provides higher interoperability, and can apply to any sort of domain application owing to the component based architecture.

The existing security management model usually considers static security policies. Thus, it will be quite difficult to deal with the changes of the environment promptly and appropriately. Thus, we propose an adaptive security management model for AAF, which could dynamically adapt security policies according to the changes of dynamic network environments. We use MAUT (Multi-Attribute Utility Theory) and Simple Heuristics to introduce contextual information to determine appropriate security policies.

It is also designed to extend its function to EAI (Enterprise Application Integration), such as legacy systems. Mid-level companies, which are difficult to develop a full-fledged RFID application due to the prototypical developing cost and/or the integrating cost to the existing legacy systems, easily construct the RFID application, and track and trace a logistics flow by utilizing EPCIS and ONS. We, therefore, can construct a general, reusable, and extensible ALE application which would be used in the diverse domains such as logistics, manufacturing, and the automation of supply chain.

4 The Context-Aware Security Service

In this section, we suggest context-aware security service model using an adaptive security level algorithm based on MAUT and Simple Heuristics.

4.1 The Adaptive Security Level Algorithm

We present the security policy algorithm that dynamically adapts the security level according to the contextual information. The information consists of the domain independent properties such as terminal types, and the domain dependent properties such as the sensitivity of information using MAUT and Simple Heuristics.

```
SecurityLevel(securityProblem)
// securityProblem: Determining security level

// Utilization of domain independent properties
   calculate SL by I end
   if SL = 0 then return SL // no use of security system

// Utilization of domain dependent properties
// Selection between MAUT and S. Heuristics
   if MAUT then SL = MAUT(X)
   if Simple Heuristics then SL = TakeTheBest(X);
   return SL; end;

MAUT(X)
// Determine total utility function by the interaction
// with the user according to MAUT
   $u(x_1,x_2,…,x_n)=k_1u_1(x_1)+k_2u_2(x_2)+… +k_nu_n(x_n)$;
// $k_i$: set of positive scaling constants for all $i$
// $x_i$: domain dependent variable, where $u_i(x^o_i)=0, u_i(x^*_i)=1$

   ask the user's preference and decide $k_i$
   for i = 1 to n
     do $u_i(x_i)$ = GetUtilFunction($x_i$);
   end
   return $u(x_1,x_2,…x_n)$; end;

GetUtilFunction($x_i$)
// Determine utility function due to users' preferences
// $x_i$ is one of domain dependent variables
   uRiskProne   : user is risk prone for $x_i$ - convex
   uRiskNeutral : user is risk neutral for $x_i$ - linear
   uRiskAverse  : user is risk averse for $x_i$ - concave
   x : arbitrary chosen from $x_i$
   h : arbitrary chosen amount
   <x+h, x-h>   : lottery from x+h to x-h

// where the lottery ($x^*$, p, $x^o$) yields a p chance at $x^*$
// and a (1-p) chance at $x^o$
   ask user to prefer <x+h, x-h> or x; // interaction
   if user prefer <x+h, x-h> then
     return uRiskProne;            // e.g. $u = b(2^{cx}-1)$
   elseif user prefer x then
     return uRiskAverse;           // e.g. $u = b\log_2(x+1)$
   else return uRiskNeutral;  end; // e.g. $u = bx$

// Take the best, ignore the rest
function TakeTheBest($u(x_1,x_2,..,x_n)$) returns SL
  inputs $u(x_1,x_2,..,x_n)$ : user's basic preferences
// if the most important preference is $x_i$, then only $x_i$
is considered to calculate SL
// the other properties except $x_i$ are ignored

  $u(x_1,x_2,..,x_n)$ is calculated by only considering the
value of $x_i$                                           return SL;
end TakeTheBest;
```

Fig. 3. Context-aware Adaptive Security Level algorithm

The variables of the algorithms are as follows:

1. Domain independent variables $I = (i_1, i_2, ..., i_n)$: data size, computing power, network type, terminal type, and so on.
2. Domain dependent variables $X = (x_1, x_2, ..., x_n)$: user attributes, system attributes.
3. Security level $SL = (0, 1, 2, ..., 5)$: The larger the number is, the stronger the strength is. If SL is 0, we can not utilize the security system.

The equations of determining U and SL in our algorithm are as follows.

$$U = \sum_{i=1}^{n} k_i u_i(x_i), \ (0 \leqq U \leqq 1)$$
$$SL = \lceil U * 10 \rceil / 2, \ (SL = 0, 1, ..., 5)$$

$u_i(x_i)$ is converted utility value for the variable x_i, and k_i is a scaling constant of each variable which is determined by security polices and security preference of user. U is the total utility value. SL is the final security level in the proposed model.

The overall algorithms for determining adaptive security level are shown in Figure 3.

4.2 Security Policy and Service Policy

Security policy plays a role in determining security level, and service policy determines parameters for security services such as access control, authentication, confidentiality, digital signature, and so on. Table 2 is a typical example of security policy, where x_{att} is the strength of the cipher, x_{auth} is the authentication method, and x_{res} is the level of protection of the resource to which the user is trying to access.

Table 2. An example of security policy

A Security Policy for *Managed Resource A*	
Security Service	Reading
Utility Function	$u(x_{att}, x_{auth}, x_{res}) = k_{att}\, u(x_{att}) + k_{auth}\, u(x_{auth}) + k_{res}\, u(x_{res})$;
Security Contexts	*comp* ≥ 200 MHz; *nType* ≥100 Kbps; *tType* = PC/PDA/Cell;
User's Preference	*uRiskProne*=$2^{2(x-1)}$; *uRiskNeutral*=*x*; *uRiskAverse*=$log_2(x+1)$;

Table 3 is an example of conversion table from the values of the security variables to the corresponding quantitative utility values. This table is used for utility analysis.

Table 3. Conversion table of each variable

utility variable	0.2	0.5	1.0
x_{att}	$\geq 10^{0.5}$	$\geq 10^{5}$	$\geq 10^{11}$
x_{auth}	One-Time Password	OTP + Certificate	OTP + Biometrics
x_{res}	Low	Medium	High

Table 4 is an example of service policy for access control where reading or writing access right is given to the user. *SL* is the lower bound of security level. Any user cannot adopt *SL* lower than 3 for writing operation. If the user is administrator and *SL* is higher than 3, then he or she can have writing permission.

Table 4. An example of access control

A Service Policy for *Managed Resource A*
If ((SL ≥ 2) **and** ((Role = administrator) **or** ((Role = user) **and** (Date = Weekdays **and** 8:00 < Time < 18:00)))) **Then** resource A can be read
If ((SL ≥ 3) **and** (Role = administrator)) **Then** resource A can be written

4.3 Dynamic Changes of Adaptive Security Level

When the security level is changed due to environmental issues such as user's preference and network's configuration, this type of change should be dynamically reflected to the current security services. Thus, we should have the mechanism to change the security services dynamically according to the modified security level.

5 Conclusion

The existing security models could not provide an appropriate security management since they have only static security management policies. In this paper, we proposed an ALE Application Framework (AAF) using context-aware adaptive security service, which could dynamically adapt policies according to the changes of diverse and dynamic network environments. The characteristics of the proposed model are as follows.

Firstly, the proposed model deals with multiple variables instead of single variable in the existing approach. Therefore, our model could provide more flexible security management in the heterogeneous network environments.

Secondly, from the proposed model, we can construct RFID application fast and efficiently through general, reusable, and extensible API due to the software reusability.

In the future, we will extend context-aware adaptive security services to full-fledged security services, such as encryption, decryption, digital signature, and non-repudiation. Also we will investigate the business application framework in more detail to leverage linkage between enterprise application and middleware or EPC related information.

References

[1] Auto-ID Center, *EPC Information Service*, White Paper, 2004.

[2] A. K. Dey, "*Providing Architectural Support for Building Context-Aware Applications*," Ph. D. Dissertation, Georgia Institute of Technology, 2000.

[3] *e-Biz Standardization White Paper*, Ministry of Commerce, Industry and Energy, Korea Institute for Electronic Commerce, 2004.

[4] EPCglobal, *Object Name Service(ONS) 1.0*, Working Draft Version , 2004.

[5] EPCglobal, *The Application Level Events (ALE) Specification, Version 1.0*, Specification, 2005.

[6] Guanling Chen, "*A survey of context-aware mobile computing research*," Dartmouth Univ.TR2000-38.

[7] R.L. Keeney and H. Raiffa, *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*, John Wiley & Sons, New York, NY, 1976.

[8] L. Martignon and U. Hoffrage, *Why Does One-Reason Decision Making Work? In Simple Heuristics That Make Us Smart*, Oxford University Press, New York, 1999, pp. 119-140.

[9] *The SSL Protocol Version 3.0*, http://wp.netscape.com/eng/ssl3/draft302.txt.

[10] W. Stallings, W. Stallings, *Cryptography and Network Security*, 3rd Ed, Prentice Hall, NJ, 2003.

[11] *The TLS Protocol Version 1.0*, http://www.ietf.org/rfc/rfc2246.txt.

[12] D. Winterfeld, von and W. Edwards, *Decision Analysis and Behavioral Research*, Cambridge, England: Cambridge University Press, 1986.

[13] *The Workflow Management Coalition*, http://www. wfmc.org.

A Study on the Standard of Software Quality Testing*

Hye-Jung Jung[1], Won-Tae Jung[2], and Hae-Sool Yang[3]

[1] Department of Information Statistics PyongTack University,
PyongTack-City, Kyonggi, 450-701, Korea
jhjung@ptuniv.ac.kr

[2] Department of Computer Information, Kyung-Moon College,
PyongTack-City, Kyonggi, 450-701, Korea
wtjung@kmc.ac.kr

[3] Graduate School of Venture, HoSeo Univ. 1603-54, Seo-cho dong,
Seo-cho gu, Seoul, 137-070, Korea
hsyang@office.hoseo.ac.kr

Abstract. There are increasing desire for software quality, in the customer and user. In general, software product quality has significant influence on developers, users of the software product. A software quality model is a very useful instrument for software quality requirement as well as software quality evaluation. ISO/IEC JTC1/SC7/WG 6 is studying for software quality testing model. ISO/IEC 9126 provides a software product quality testing model. Many testing center use the ISO/IEC 9126 quality model for software testing. In these days, ISO/IEC JTC1/SC7/WG 6 is developing ISO/IEC 25000 SQuaRE (Software Quality Requirements and Evaluation) series of international standards. ISO/IEC JTC1/SC7/WG 6 studied to start to develop a new Quality model(25010) as a revision of ISO/IEC 9126-1 Quality Model. Also, ISO/IEC JTC1/SC7/WG 6 is studying to revise 9126-2, 9126-3, 9126-4, as a part of the SQuaRE series of International Standards(IS) ISO/IEC 25022, 25023, 25024. The purpose of this paper is to study on the International Standards for software quality testing. ISO/IEC JTC1/SC7/WG 6 is studying for SQuaRE project in these days. SQuaRE project includes International Standards on software quality model and software quality measures, as well as on software quality requirements and software quality evaluation. SQuaRE replaces the current ISO/IEC 9126 series and the 14598 series. In this paper, we survey the SQuaRE project detailed for software quality testing.

1 Introduction

There is an increasing needs for software that matches real user needs in a working enviroments. ISO/IEC JTC1/SC7/WG 6 developed for ISO/IEC 9126 international standards software quality model. ISO/IEC 9126 consists of the following parts under the title Software Engineering-Software product quality.(Part 1:Quality model, Part 2:External Metrics, Part 3:Internal Metrics, Part 4:Quality in use Metrics). This Internal

* This research was supported by University IT Research Center Project.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1052 – 1059, 2006.

Technical Report 9126-2 External Metrics contains an explanation of how to use software quality metrics(Functionality, Reliability, Usability, Efficiency, Maintainability, Portability), a basic set of metrics for sub-characteristic.

In these days, we use ISO/IEC 9126 for software quality testing in country. Also, SQuaRE series replaces the current ISO/IEC 9126 series and the ISO/IEC 14598 series.

SQuaRE series of standards consists of the following divisions under the general title Software product Quality Requirements and Evaluation.(Quality Management Division(2500n), Quality Model Division(2501n), Quality Measurement Division(2502n), Quality Requirement Division(2503n), and Quality Evaluation Division(2504n)).

ISO/IEC 9126 has the 4 parts. Where, internal quality is measured by the static properties of the code, external is measured by the dynamic properties of the code, and quality in use is measured by the extent to which the software meets the needs of the user in the working environment. Software quality can be measured internally or externally. Software reliability can be measured externally by observing the number of failures, number of fault in a given time during a trial of the software testing. Software reliability metrics can not measured because failure datas doesn't obtain.

We have to find the measuring method software quality. Software quality measured the 6 characters.(Functionality, Reliability, Usability, Efficiency, Maintainability, Portability).

The rest of this paper is arranged as follows: Chapter 2 describes the commonly used software quality testing model and history.

Chapter 3, we introduce the ISO/IEC 25000 series. Also, we introduced the measurement primitive and propose the software reliability metric in country.

Chapter 4, we introduce the conclusion and future study in software testing model.

2 History of the Software Quality Testing Model

The software testing model ISO/IEC 9126 consist of the 6 characters.(Functionality, Reliability, Usability, Efficiency, Maintainability, Portaility). Functionality is the capability of the software to provide functions which meet stated and implied needs when the software is used under specified condition.. Reliability is measured the software to maintain its level of performance when used under specified conditions.

Software Usability is the capability of the software to be understood, learned, used and liked by the user, when used specified conditions.

Software Efficiency is the capability of the software to provide the required performance, relative to the amount of resources used, under stated conditions.

Software maintainability is the capability of the software to be modified. Modification may include corrections, improvements and functional specifications.

Software portability is the capability of software to be transferred from one environment to another.

Software quality testing model is a very useful tool for quality requirement and quality evaluation.

ISO/IEC JTC1/SC7/WG6 is studying ISO/IEC 25000 SQuaRE series. We have to fine software quality testing model for our country. Qur country has the software quality standards of testing model in TTA, KICS, KS X . We have to study software quality testing model to establishment by law for international standards.

For example ISO/IEC 25000 series studied ISO/IEC JTC1/SC7/WG 6 as follows;

ISO/IEC 25000: Software and System engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Guide to SQuaRE.

NP	WD	CD	2CD	FCD	FDIS
		2002.10	2003.04	2004.05	2005.10

ISO/IEC 25001: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Planning and management.

NP	WD	CD	DIS	Publish	KS
		2005.05			

ISO/IEC 25010: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Quality model.

NP	WD	CD	DIS	Publish	KS

ISO/IEC 25020: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Measurement reference model and guide.

NP	WD	CD	2CD	3CD	KS
		2003.04	2004.04	2004.09	

ISO/IEC 25021: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Measurement primitives.

NP	WD	CD	PDTR	Publish	KS
	2004.04	2004.04	2005.05		

ISO/IEC 25022: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Measurement of internal quality.

NP	WD	CD	DIS	Publish	KS

ISO/IEC 25023: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Measurement of external quality.

NP	WD	CD	DIS	Publish	KS

ISO/IEC 25024: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Measurement of quality in use.

NP	WD	CD	DIS	Publish	KS

ISO/IEC 25030: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Quality requirements.

NP	WD	CD	2CD	Publish	KS
	2002.11	2003.05	2004.04		

ISO/IEC 25040: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Evaluation reference model and guide.

NP	WD	CD	DIS	Publish	KS

ISO/IEC 25041: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Evaluation modules.

NP	WD	CD	DIS	Publish	KS

ISO/IEC 25042: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Evaluation process for developers.

NP	WD	CD	DIS	Publish	KS

ISO/IEC 25043: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Evaluation process for acquirers.

NP	WD	CD	DIS	Publish	KS

ISO/IEC 25044: Software engineering: Software product Quality Requirements and Evaluation(SQuaRE) - Evaluation process for evaluators.

NP	WD	CD	DIS	Publish	KS

ISO/IEC 25022, 25023, 25024, 25041, 25042, 25043, 25044 have to study in country professional of the software quality.

Especially ISO/IEC 2502n, we study about criteria for selecting software quality measures, deconstrating predictive validity and assessing measurement reliability, and software quality metrics. ISO/IEC 2504n series make up 5 parts.

25041 defines the structure and content of the documentation to be used to describe an Evaluation Module. 25042 provides requirements and recommendations for the practical implementation of software product evaluation when the evaluation is conducted in parallel with the development. 25043 contains requirements, recommendations and guidelines for the systematic measurement, assessment and evaluation of software product quality during acquisition of "off-the-shelf" software products, custom software products, or modifications to existing software products. 25044 provides requirements and recommendations for the practical implementation of software product evaluation, when several parties need to understand, accept and trust evaluation results.

3 Software Reliability Metrics

The measurement primitives needed to constract the internal quality, external quality and quality in use measures shall be defined. It is a measurement primitive table as follows.

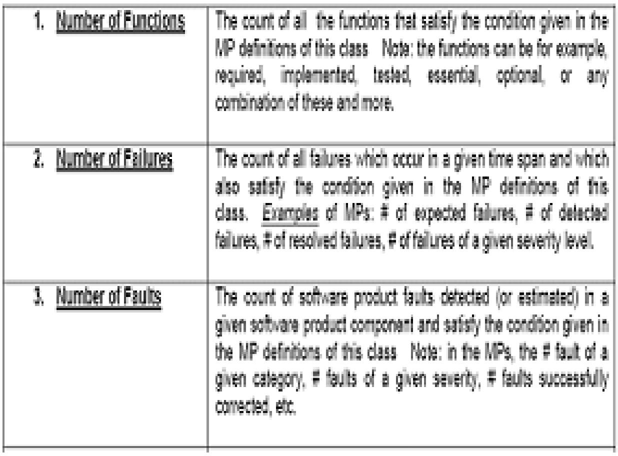
Measurement primitive studied the ISO/IEC JTC1/SC7/WG6.

Measurement primitive refer to the user, developer, tester in software.quality testing model.

For example, software reliability measure the probability of the software's function under the condition.

This time, we use the number of failure in measurement primitive.

Table 1. Measurement Primitive



1. Number of Functions	The count of all the functions that satisfy the condition given in the MP definitions of this class Note: the functions can be for example, required, implemented, tested, essential, optional, or any combination of these and more.
2. Number of Failures	The count of all failures which occur in a given time span and which also satisfy the condition given in the MP definitions of this class. Examples of MPs: # of expected failures, # of detected failures, # of resolved failures, # of failures of a given severity level.
3. Number of Faults	The count of software product faults detected (or estimated) in a given software product component and satisfy the condition given in the MP definitions of this class Note: in the MPs, the # fault of a given category, # faults of a given severity, # faults successfully corrected, etc.

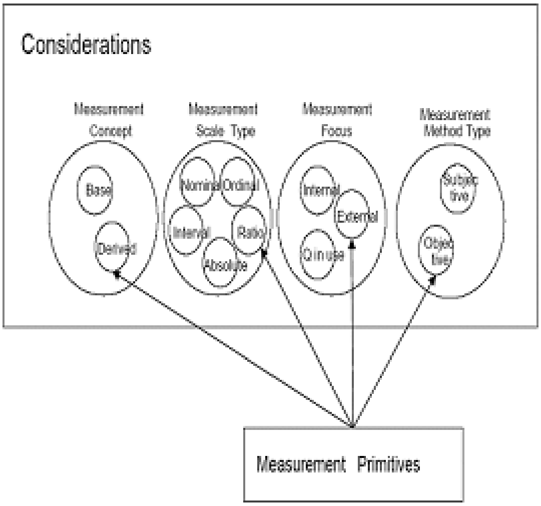


Fig. 1. Measurement Primitive Considerations

Base measures Measurement Primitives measure an attribute of resource, process, product and product use that does not depend upon a measure of any other attribute. Derived Measures Measurement Primitives measure an attribute of the resource, process, product, product use that depends upon a measure of any other attribute. Internal Measure Measurement Primitives are measuring internal attributes of a software product itself. The number of lines of code, complexity measures and the number of faults found in a walk through are all internal measures made on the product itself.

Software reliability is difficult to calculation. Software reliability used the software reliability growth model.. So, many researcher study the software reliability growth model.

For example, researchers have studied the modeling of software reliability. The Jelinski-Moranda(1972)[9] model for software reliability growth model(SRGM) is one of the most commonly cited models. The main property of the model is that the intensity between two consecutive failure is constant. And the distribution of inter-failure time T_i at the i-th testing stage is given by

$$f(t_i)= (N-i+1)\ \phi\ \exp((-N-i+1)\phi t_i)$$

Where the parameter $\lambda_I =(N-i+1)\phi$ is a failure rate, and N is the number of latent software errors before the testing starts, and ϕ is the failure intensity contributed by each failure. The parameter ϕ and N in Jelinski-Moranda model was estimated by maximizing the likelihood function.

The software reliability growth model suggested by Littlewood and Verrall(1973)[10] is perhaps the most well known Bayesian model. The Littlewood-Verrall model assumes that interfailure times are exponential distributed random variable with the density function.

$$f(t_i| \lambda_i)= \lambda_i \exp(-\lambda_i t_i)$$

where λ_i is an unknown parameter whose uncertainty is due to the randomness of the testing and the random location of the software errors. And the probability density function of λ_i is

$$f(\lambda_i \mid \alpha, \varphi(i))= \frac{(\varphi(i)^{\alpha} \lambda_i^{\alpha-1} \exp(-\varphi(i)\lambda_i)}{(\Gamma(\alpha))}$$

where $\varphi(i)$ is depending on the number of detected error. Usually, $\varphi(i)$ describes the quality of the test and it is a monotone increasing function of i.

Langberg and Singpurwalla(1985)[12] presented a shock model interpretation of software failure and justified the Jelinski-Moranda model.

In the Langberg and Singpurwalla Bayesian model the parameters in the Jelinski-Moranda model are treated as random variables.

Nayak(1986)[13] proposed a model for incorporating dependence among the detection times of software reliability measures. Many software reliability model have been studied . But they treated with the case of software reliability growth model for single error debugging at each testing stage until now. Jung[20] studied the software reliability model with the multiple errors debugging.

Jung[20] introduce this distribution as a Gamma-Lomax Software Reliability Growth Model which is an extension case of the multivariate Lomax distribution of Nayak[13].

Many software research proposal the software reliability model.

<Table 2> is the software reliability metric. We have to calculate the software reliability metrics for using the software reliability model. So we study the mathmetical

testing method for software reliability metrics. We study measurement primitive using experience of software testing.

Table 2. Reliability Table

Estimated failure density		Number of failure Total number of failure	
Measure Item	NPFI	Estimated the number of failure	
	NAFI	Actured number of failure	
	SIZE	Product size	
Calculation	X=ABS(NPFI-NAFI)/SIZE		
Result	0 ≤Estimated failure intensity		
Value		Improvement	

4 Concluding Remarks

In this paper, we expect the result of study following below.

Technical Effect is that software quality testing is technology which can introduce quality enhancement of software and strategic technology which accept international standard. So, we can test the software quality evaluation exactly.

Industrial and economical effect is that high value of software industry and economical effect is very important. Software quality testing is essential technology for software industry. With this standards, the quality of software produced by domestic software developers can be improved because the quality of software is evaluated objectively. Also, the market for software could be vitalized.

Social Effect is that we can solve the software quality evaluation of information society by using this study result.

So, we are proposal the practical using. We survey and study for the tendency of the software quality evaluation in inside and outside of the country. We can use in establishment of the standard of software quality evaluation. Also, we propose the using these study result in software quality tester training. Also we have expected effect in this paper.

First, establishment of the plan of software quality evaluation according to international standard.

Second, we use the establishment of standard in software quality evaluation.

Third, acquisition of stability for related market by quality enhancement of domestic software. Forth, we have opposition method and spread method in international standard of software quality evaluation.

Recently, software quality assurance activities for the development procedure concerned with the software development project have been highly concerned as well as the software product itself. However the measurement of the software reliability is very difficult. We suggest that the software reliability model should be applied for the standardization and the software quality measurement activities of the software product.

We are going to study deep learning measurement primitive for software testing.

References

1. ISO/IEC 9126 "Information Technology-Software Quality Characteristics and metrics-Part 1,2,3.
2. ISO/IEC14598 "Information Technology-Software Product Evaluation-Part 1~6.
3. ISO/IEC12119 "Information Technology-Software Package-Quality requirement and testing".
4. A.L. Goel, and K. Okumoto, "Time-dependent error detection rate model for software reliability and other performance measures", IEEE Trans. Reliability, R-28, pp206-211, 1979.
5. A.L. Goel, "Software reliability model : assumptions, limitations, and applicability", IEEE Trans Software Eng, SE-11, pp 1411-1423, 1985.
6. L.J. Bain, 'Statistical Analysis of Reliability and Life-Testing Models', MARCEL, DEKKER, 1991.
7. Catuneanu, V.M. et al, "Optimal software release policies using SLUMT", Microelectronics and Reliability, 28, pp. 547-549, 1988.
8. Catuneanu, V.M. et al, " Optimal software release time with learning rate and testing effort", IEEE Trans. Reliability, 1991.
9. Z. Jelinski, and P.B. Moranda, "Software reliability research, in Statistical Computer Performance Evaluation", Ed. W. Freiberger, Academic Press, New York, pp.465-497, 1972 .
10. B. Littlewood, and J. L. Verrall, " A Bayesian reliability growth model for computer software", Applied Statistics, 22, pp.332-346, 1973.
11. K. Okumoto, and A.L. Goel, " Optimum release time for software systems based on reliability and cost criteria", J. Systems and Software, 1, pp.315-318, 1980.
12. N. Langberg, and N.D. Singpurwalla, "A Unification of some software reliability models", SIAM J. Scientific and Statistical Computation,6, pp.781-790, 1985.
13. T. K. Nayak, "Software Reliability:Statistical Modeling and Estimation", IEEE Trans. On Reliability, 35, pp.566-570, 1986.
14. M. L. Shooman, "Software Reliability : measurement and models", Proc. Ann. Reliability and Maintainability Symp, pp.485-491 , 1975.
15. S. Yamada, and S. Osaki, "Cost-reliability optimal release policies for software systems", IEEE Trans. Reliability, R-34, pp422-424, 1985.
16. S. Yamada, and S. Osaki, " Optimal software release polices for nonhomogeneous software error detection rate model", Microelectronics and Reliability, 26, pp.691-702, 1986.
17. S. Yamada, and S. Osaki, "Cost-reliability optimal release policies for software systems", IEEE Trans. Reliability, R-34, pp422-424, 1985.
18. S. Yamada, and S. Osaki, " Optimal software release polices for nonhomogeneous software error detection rate model", Microelectronics and Reliability, 26, pp.691-702, 1986.
19. A.O.C. Elegbede, C.Chu, K.H. Adjallah, & F. Yalaoui, "Reliability Allocation Through Cost Minimization", IEEE Trans on Reliability, 52, 1, pp.96-105, 2003.
20. H. J. Jung . "Software Reliability Growth Modeling and Estimation for Multiple Errors Debugging", Kyungpook National University, 1994.

Scene Change Detection Using the Weighted Chi-Test and Automatic Threshold Decision Algorithm[*]

Kyong-Cheol Ko, Oh-Hyung Kang, Chang-Woo Lee,
Ki-Hong Park, and Yang-Won Rhee

Department of Computer Science, Kunsan National University,
68, Miryong-dong, Kunsan, Chonbuk 573-701, South Korea
{roadkkc, ohkang, leecw, spacepark, ywrhee}@kunsan.ac.kr

Abstract. This paper presents a method for detecting scene changes in video sequences, in which the χ^2-test is slightly modified by imposing weights according to NTSC standard. To automatically determine threshold values for scene change detection, the proposed method utilizes the frame differences that are obtained by the weighted χ^2-test. In the first step, the mean of the difference values is calculated, and then, we subtract the mean difference value from each difference value. In the next steps, the same process is performed on the difference values, mean-subtracted frame differences, until the stopping criterion is satisfied. Finally, the threshold value for scene change detection is determined by the proposed automatic threshold decision algorithm. The proposed method is tested on various video sources and, in the experimental results, it is shown that the proposed method is reliably detects scene changes.

1 Introduction

For the processing of a huge amount of video data, it is necessary to develop fast and efficient techniques in indexing, browsing and retrieval of videos [1]. Video segmentation is the first step to establish video database systems and it is an essential work to segment video sequence into shots where each shot represents a sequence of frames having the same contents [2].

There are two kinds of scene change, abrupt and gradual. The abrupt scene change is the change of scene with cut of camera and the gradual scene change is that of scene with camera action for fade-in, fade-out, and dissolves [3], [4]. It is generally known that abrupt scene change detection is easier than gradual one. For detecting scene changes, thresholds have to be pre-assigned. This is the major problem of the scene change detection, and it is difficult to specify the correct threshold that determines the performance of scene change detection [5].

To segment a video sequence into scenes, a number of scene change detection algorithms have been reported in the literatures. In general, these algorithms can be

[*] This research was supported by the Program for the Training of Graduate Students in Regional Innovation which was conducted by the Ministry of Commerce, Industry and Energy of the Korean Government.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1060 – 1069, 2006.

categorized into four parts: Pixel-based algorithms [6], Histogram-based algorithms [7], Block-based algorithms [8], [9] and Clustering-based algorithms [10], [11].

Pixel-based algorithms compare the pixels of two adjacent frames across the same location. Then, a scene change is declared if inter-frame difference (the sum of pixel by pixel differences across the same location) between consecutive frames exceeds the pre-assigned threshold. In histogram-based algorithms, the difference of the intensity or color histogram is used as dissimilarity measure between two frames. Histogram-based methods provide a better tradeoff between accuracy and speed, and its performance is good for the case of abrupt scene changes such as cuts. The best performance is obtained by χ^2-test [7]. For block-based algorithms, local attributes are used to measure the difference of two frames so that the effect of noise or camera flash can be reduced. A scene change is detected if the number of blocks that the differences of corresponding blocks in two frames are greater than the pre-assigned threshold exceeds a given lower bound. The pre-assigned threshold is also the case with histogram-based algorithms and block-based algorithms. Clustering techniques are used to categorize all the frames in a video sequence into several clusters (e.g. k-clusters in K-means clustering algorithm), in which a cluster represents a scene.

All scene change detection algorithms are based on a pre-assigned threshold. If the threshold is too low, many key frames are extracted so that a video sequence is over-segmented. On the contrary, for a high threshold, many key frames may be missed, resulting in under-segmentation. Above all, a threshold that is appropriate for a variety of video types has to be determined previously. That is, a threshold for a video sequence is not appropriate for the others and it is not guaranteed that the threshold for one type of video data will not yield acceptable results for other types of inputs.

To solve the above mentioned problem, this paper proposes a method to determine thresholds that is adaptive for a variety of input video sequences. For robust scene change detection, we slightly modify the χ^2-test by imposing different weights on each channel of the RGB color space. We refer this χ^2-test as the weighted χ^2-test. To determine thresholds automatically according to an input video type, the proposed method utilizes the frame differences that are obtained by performing the weighted χ^2-test.

2 The Proposed Scene Change Detection Method

2.1 The Weighted χ^2-Test

In this paper, we calculate the frame differences from the weighted χ^2–test which combined the color histogram with χ^2–test. The weighted χ^2-test can subdivide the difference values of individual color channels by calculating the color intensities according to NTSC standard. The weighted χ^2–test formula ($d_{w\chi^2}$) is defined as

$$\begin{aligned} d_{w\chi^2}(f_i, f_j) = \frac{1}{3N} \sum_{K=0}^{N-1} \Bigg(& \frac{(H_i^r(k) - H_j^r(k))^2}{\max(H_i^r(k), H_j^r(k))} \times \alpha + \frac{(H_i^g(k) - H_j^g(k))^2}{\max(H_i^g(k), H_j^g(k))} \times \beta \\ & + \frac{(H_i^b(k) - H_j^b(k))^2}{\max(H_i^b(k), H_j^b(k))} \times \gamma \Bigg) \end{aligned} \quad (1)$$

where N is the number of bins, and $H_i^r(k)$, $H_i^g(k)$, and $H_i^b(k)$ are the bin values of the histogram of i'th frame in red, green, and blue color channels, respectively. The α, β, and γ are constants and, according to NTSC standard, we set these constants to 0.299, 0.587, and 0.114, respectively.

Figure 1 shows the frame differences on a test video sequence, which are calculated by the χ^2–test and the weighted χ^2–test. As shown in figure 1, the weighted χ^2–test has an advantage than the χ^2–test in that the possibility of automatic threshold decision is high.

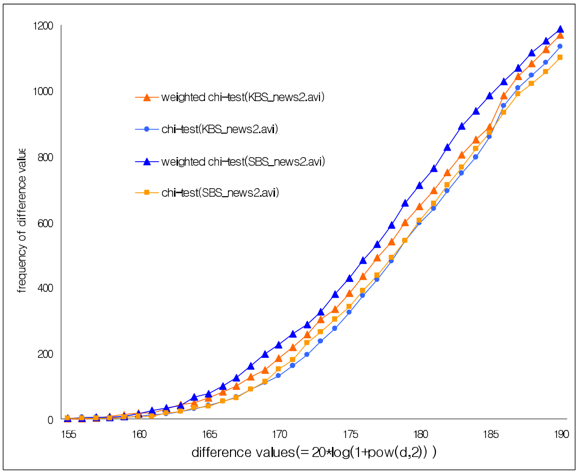


Fig. 1. Comparison of the χ^2–test and the weighted χ^2–test on an input video sequence

2.2 Automatic Threshold Decision Algorithm Using the Means and Standard Deviation Values from Video Sequence

The difference values obtained from the weighted χ^2–test is the basic data to extract the representative frames which occurred during the scene change. Thus, obtained data can be used for the feature extraction and it can be used for the detection of scene change. Pseudo code-1 shows the procedural calculation course of the mean and the standard-deviation from the extracted difference values to decide the most proper threshold from video sequence.

Pseudo code 1. Automatic threshold decision algorithm

Step 1: Calculate the total difference values and 1_th mean value from given video sequences

```
for( i=1; i<=n; i++){ // n = total number of frames
    Difference values $X = \{x_1, x_2, x_3, \ldots x_n\}$;
    // X is calculated using the weighted •$^2$-test from the consecutive frames ($f_i$, $f_j$) ;}
    First mean value m = $1/n \times \sum_{i=1}^{n} x_i$ ;

    First standard deviation value $\sigma$ = $\sqrt{1/n \times \sum_{i=1}^{n} (x_i - m)^2}$ ;
```

Step 2: Calculate the t_th mean and standard deviation value

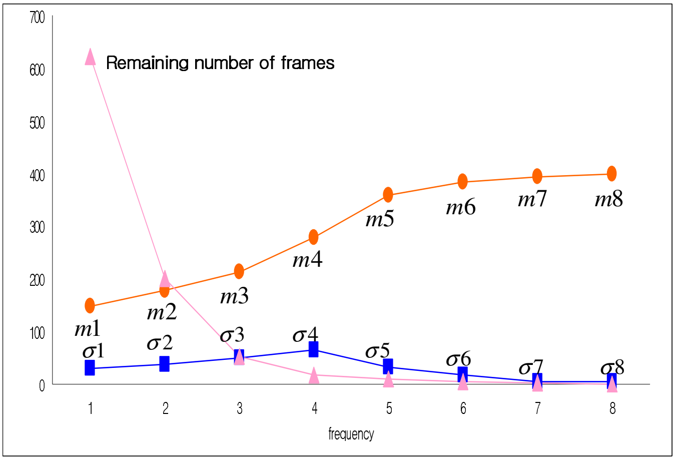
```
 for( i=1; i<= n; i++) {
   if($x_i$ > $m^{t-1}$)
     $X^t$ = { $x_i$}; // $X^t = \{x_i, \dots x_p\}$ and $(i \le p \le n)$
   else $x_i$ is eliminated; }
   t_th mean value $m^t$ = $1/p \times \sum_p x_p$ ;

   t_th standard deviation value $\sigma^t$ = $\sqrt{1/k \times \sum_p (x_p - m^t)^2}$ ;}

if($\sigma^t$ > $\sigma^{t-1}$) goto Step 2;
else  goto step 3;
```

Step 3: Decide the threshold value

```
// automatic threshold value is decided
      $th$ = $m^t$ ; //threshold representative value
      $f_n$ = $p$ ; //number of representative frames
// confidence interval values are calculated to measure the
sensitivity of estimated thresholds.
// $\alpha$ (%) confidence interval ($\alpha$ (95%) = 1.96)
   $val$ = $\alpha \times \sigma^t / \sqrt{f_n}$ ; // confidence calculation
   // calculation of max and min means based on confidence interval
using the $m^t$ .
   $th_{MAX}$ = $m^t$ + $val$ ; // max_threshold confidence interval
   $th_{MIN}$ = $m^t$ - $val$ ; // min_threshold confidence interval
   // number of frames for the max and min mean values
    for(i=1; i<=n; i++) {
      if( $x_i$ > $th_{MAX}$ ) $f_{\max}$ ++;
if( $x_i$ > $th_{MIN}$ ) $f_{\min}$ ++; }
```

The mean and standard-deviation values extracted from the difference values are the meaningful data to decide the automatic threshold. The mean is the essential data to decide the threshold from the extracted difference values and standard deviation is the reference data to automatically decide the best of threshold from the calculated means. The automatic decision of threshold value is based on the calculated means and standard deviation values over an entire difference values. The proposed automatic threshold decision algorithm can decide more adaptive threshold according the video sequence type.

Figure 2 shows a distribution of means and standard deviation values after the means exclusion according to the algorithm which is sequentially calculated by the proposed Pseudo code1. All possible means and standard deviation values are calculated and the remaining numbers of frames are checked.



Frequency(k)	1	2	3	4	5	6	7	8
Means(m_k)	149.1	177.0	212.4	277.4	359.7	383.1	394.8	398.7
Standard-deviation (σ_k)	30.5	38.5	51.0	64.4	32.6	16.4	5.9	6.2
Number of frames	621	201	52	18	10	6	3	1

Fig. 2. Distribution graph of means, standard deviation and remaining number of frames

As shown in Figure 2, we can estimate the mean values (m1~m4) as the threshold according to step 3 course of the proposed algorithm, and 4-th means value is the proper threshold because the standard deviation value is bigger than the previous values.

So the estimated threshold value (m4) can detect 18 numbers of representative frames and it shows all possible abrupt scene change (12) are detected and the rest frames (6) is a gradual scene changes. If the mean is too low, many frames are extracted so that a representative frames is over segmented. On the contrary, for high means, many representative frames may be missed. So the threshold selection is a difficult problem.

Figure 3 shows a normal distribution using the calculated means and standard-deviation values which was calculated in Figure 2. It shows a more widely normal distribution graph in 4-th means and standard-deviation values. This paper propose the automatic threshold decision algorithm and the threshold can be detected from the means and standard deviation values which is sequentially calculated by the proposed Pseudo code1.

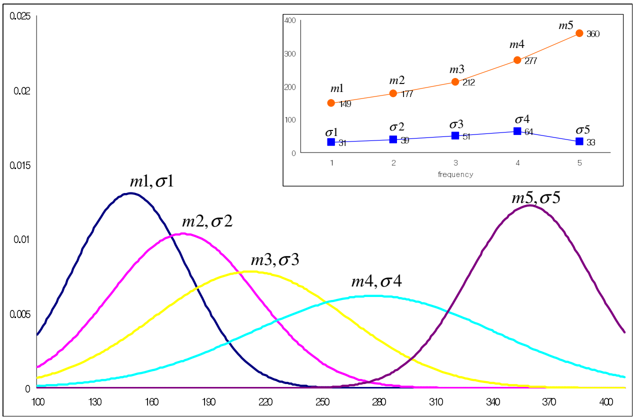


Fig. 3. Normal distribution graph using the mean and standard deviation values

Figure 4 shows the distribution graph of remaining number of frames after means exclusion.

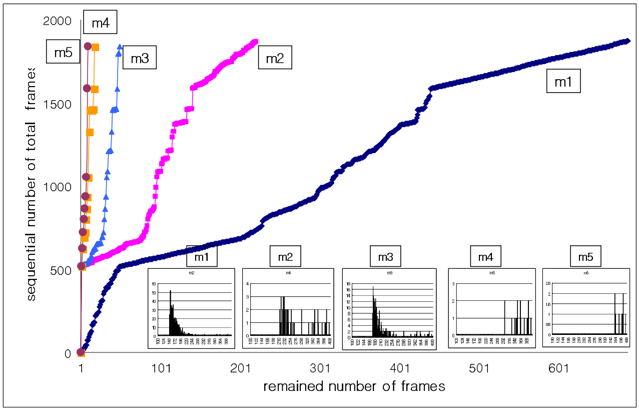


Fig. 4. Distribution of remaining number of frames by the proposed algorithm

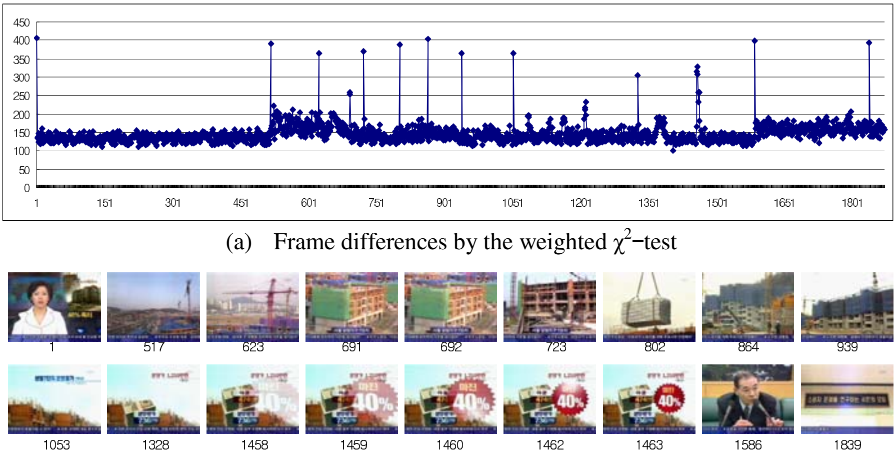
In our experimental results, the most proper threshold can be detected using the maximum standard-deviation values because of the normally distributed frames. Thus, if the standard deviation has maximum value, the corresponding means are selected to the threshold.

3 Experimental Results

We evaluate the performance of our proposed method with DirectX 8.1 SDK, MS-Visual C++ 6.0 on Windows XP. The proposed method has been tested on several video sequences such as news videos that a lot of scene changes occurs, as shown in table 1. Each video sequences has the various types digitized in 320x240 resolution at 20 frames/sec.

Table 1. Video sequence types used in the experiment

Sequences	# of frames	# of predefined scene changes (A)		
		Abrupt	Gradual	Total
News1	397	5	0	5
News2	1857	17	0	16
News3	898	10	1	11
News4	670	10	0	10
News5	1871	12	2	14
Adv_1	697	23	3	26
Adv_2	720	4	1	5
Adv_3	652	6	2	8
Adv_4	682	7	2	9
Adv_5	592	9	4	13



(a) Frame differences by the weighted χ^2–test

(b) Extracted representative frames

	1			2			3			4			5		
	95% (−)	m1	95% (+)	95% (−)	m2	95% (+)	95% (−)	m3	95% (+)	95% (−)	m4	95% (+)	95% (−)	m5	95% (+)
m	144.8	146	147.2	165.3	167.8	170.2	187.9	194.2	200.5	233.1	254	275	314.9	340.6	366.3
σ	26.66			33.00			47.78			74.84			55.57		
f_n	730	687	644	254	221	195	64	49	35	19	18	14	12	10	7

(c) Distribution of mean and standard deviation values calculated from given video sequence

Fig. 5. Example of the scene change detection using the proposed algorithm

Figure 5 shows an example of scene change detection, in which frame difference are calculated by the weighted χ^2−test and the extracted keyframes are the representative frames of scene change in video sequence. Distribution of mean and standard deviation value is calculated with the proposed automatic threshold decision algorithm.

The performances of scene-change-detection algorithms are usually expressed in terms of recall and precision. The recall parameter defined the percentage of true detection with respect to the overall events (scene changes) in the video sequences. Similarly, the precision is the percentage of correct detection with respect to the overall declared event. The recall and precision are defined as

$$\text{Recall} = \frac{N_c}{N_c + N_m} \times 100\% \text{, and } \text{Precision} = \frac{N_c}{N_c + N_f} \times 100\% \tag{2}$$

where N_c: number of correct detection; N_m: number of miss; N_f: number of false detection; $N_c + N_m$: number of the existing events; $N_c + N_f$: number of overall declaration.

Table 2 show the automatic decided thresholds and detected number of frames according to the automatic threshold decision algorithm. The means of confidence interval (95%) has the maximum value th$_{\text{max}}$ (= 95%(+)) and minimum value th$_{\text{min}}$ (=95%(-)).

Table 2. Automatic decided threshold and detected number of frames

Sequences	Automatic decided threshold			Detected number of frames(B)		
	th$_{\text{max}}$	threshold	th$_{\text{min}}$	th$_{\text{max}}$	threshold	th$_{\text{min}}$
News1	203.73	228.19	252.64	14	12	7
News2	259.88	277.41	294.94	20	18	18
News3	271.54	302.04	332.54	11	10	10
News4	247.31	261.07	274.84	27	16	11
News5	233.07	254.03	274.99	19	18	14
Adv_1	281.35	290.82	300.29	45	38	34
Adv_2	180.65	196.19	211.73	19	10	10
Adv_3	216.16	232.14	248.12	27	16	11
Adv_4	245.34	260.33	275.33	15	13	10
Adv_5	250.51	259.42	268.34	38	33	29

Means of confidence interval are calculated to measure the sensitivity of decided thresholds. The difference of detected number of frames between mean of decided threshold and mean of confidence interval value (th$_{\text{max}}$, th$_{\text{min}}$) will be adequate when it has small value.

Table 3 shows the correctness and detection rate of detected frames with the scope of threshold. We show the statistics of our experimental results in Table 3.

Table 3. Distribution of the detected rate, recall and precision

Sequences	Detected rate ((B/A) * 100)			Recall(%)			Precision(%)		
	th_{max}	threshold	th_{min}	th_{max}	threshold	th_{min}	th_{max}	threshold	th_{min}
News1	280	240	140	100	100	100	61	63	78
News2	118	106	106	100	100	100	87	95	95
News3	100	90	90	92	91	91	92	100	100
News4	270	160	110	100	100	100	61	73	92
News5	136	129	100	100	100	93	79	86	100
Adv_1	173	146	131	100	100	100	70	76	81
Adv_2	380	200	200	100	100	100	58	67	67
Adv_3	338	200	138	100	100	100	59	67	79
Adv_4	167	144	111	100	100	91	71	76	91
Adv_5	292	254	223	100	100	100	66	62	64
Average	225.4	166.9	134.9	99.2	99.1	97.5	70.4	76.5	84.7

For the propose scene change detection algorithm, the reported recall is about 99.1% and the detected rate is about 166.9%. Most of the scene changes are detected. The missed scene changes are very ambiguous due to the slow camera movement.

The purpose of proposed algorithm in not to detect the correct scene change frames but to detect all possible scene change frames from video sequence. And also the threshold is not fixed or given, so it is automatically decided by automatic threshold algorithm to all video sequence. Experimental result shows the proposed algorithm is effective and can be adapted to video indexing system.

4 Conclusion

This paper proposed a method for scene change detection using the weighted χ^2–test and the automatic threshold decision algorithm. In experimental results, we showed that the proposed algorithm effectively calculated the frame differences and extracted scene changes with adaptive thresholds to each input video. The proposed weighted χ^2– test showed the better performance than the previous χ^2– test. As a result, the proposed algorithm could detect all possible scene changes like this, recall is about 99.1% and the detected rate is about 166.9%. In addition, we showed that the frame differences by the proposed method are applicable for determining a threshold value that is appropriate to a given input video.

This paper focused on abrupt scene changes, so, if video sequences have gradual changes, flashlights, or lightening effects, the automatically decided thresholds will not be appropriate to detect scene changes correctly. Thus, we are considering the condensation of the frame differences for the further research.

References

1. Huang C. L. and Liao B. Y.: A Robust Scene Change Detection Method for Video Segmentation, IEEE Trans on CSVT, Vol. 11, No. 12, December (2001) 1281-1288.
2. Gargi U., Kasturi R., and Strayer S. H.: Performance Characterization of Video Shot Change Detection Methods, IEEE Trans on CSVT, Vol. 10, No. 1, February (2000) 0001-0013.
3. Zhang H., Kankamhalli A. and Smoliar S.: Automatic partitioning of full-motion video, ACM Multimedia Systems, New York: ACM Press, Vol. 1, (1993) 10-28.
4. Dailianas A., Allen R. B., England P.: Comparison of Automatic Video Segmentation Algorithms, Large Commercial Media Delivery Systems, Proc. SPIE 2615, Oct. (1995) 2-16.
5. Rainer Lienhart: Comparison of Automatic Shot Boundary Detection Algorithms, Storage and Retrieval for Still Image and Video Databases VII, Proc. SPIE 3656-29, (1999).
6. Nagasaka A. and Tanaka Y.: Automatic video indexing and full-video search for object appearances, Visual Database Syst. II, (1992) 113-127.
7. Sethi I. K. and Patel N.: A statistical approach to scene change detection, SPIE, vol. 2420, (1995) 329-338.
8. Ekin A., Tekalp A.M. and Mehrotra R.: Automatic soccer video analysis and summarization, IEEE Trans. on Image Porcessing, vol. 12, no. 7, July (2003) 796-807.
9. Pengwei Hao and Ying Chen: Co-Histogram and Its Application in Video Analysis, ICME., Vol. 3, June (2004) 1543-1546.
10. Joshi A., Auephanwiriyakul S. and Krishnapuram R.: On Fuzzy clustering and Content Based Access to Networked Video Database, IEEE conference, Eighth International workshop on Continuous-Media Databases and Applications, (1998) 42-49.
11. Hanjalic A., Zhang J.:, An Integrated Scheme for Automated Video Abstraction Based on Unsupervised Clusterdity Analysis, IEEE Transactions on Circuits and Systems for Video Technology 9, (1999) 1280-1289.

Design Opportunity Tree for Schedule Management and Evaluation by COQUALMO*

Eun Ser Lee[1] and Sang Ho Lee[2]

[1] Information & Media Technology Institute, Soongsil University
eslee1@ssu.ac.kr
[2] College of Information Science School of Computing, Soongsil University
shlee@comp.ssu.ac.kr

Abstract. Project planning is often the most difficult part of project management. Relationship of jobs, risk items and human resources have been used successfully for schedule and project progress. There are many defects that cause the schedule and progress management problems during software development. This paper designs the opportunity tree framework that remove and manage the schedule and quality problems as well. For the similar projects, we can estimate defects and prepare to solve them by using domain expert knowledge and the opportunity tree framework, which can greatly improve the software process.

Keywords: Defect Management, Defect cause prioritization, COQUALMO, Defect reduction, Software Process Improvement, Quality, Project Schedule Management, Opportunity Tree.

1 Introduction

Setting sensible goals for process improvement requires an understanding of the difference between immature and mature software organizations. In an immature software organization, software processes are generally improvised by practitioners and their management during the course of the project. Even if a software process has been specified, it is not rigorously followed or enforced. The immature software organization is reactionary, and managers are usually focused on solving immediate crises (better known as fire fighting). Schedules and budgets are routinely exceeded because they are not based on realistic estimates. When hard deadlines are imposed, product functionality and quality are often compromised to meet the schedule[2][3][4].

Errors that are found during examination and testing offer an insight on the efficiency of each activity of the matrix. We infer quality from such factors, and this quality is a main factor that determines the success of a project together with cost and time schedule. Software has various quality-related characteristics. Moreover, there are various international standards for quality[1]. This paper identifies defects to produce a reliable project planning, project progress management and analyzes the

* This work was supported by the Soongsil University Research Fund.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1070 – 1080, 2006.

relationship among jobs. Also, this paper is intended to develop the relationship between defects of schedule and their causes to introduce. Therefore, using schedule defect causes, we understand the associated relationship between schedule risk items and design Opportunity Tree to manage the defects.

2 Related Work

2.1 Introduce of Defect

A defect is an important diagnostic type representing the process and product. Since a defect is closely related to the quality of software, defect data is more important than a man-month estimation in several ways [5][6].

Furthermore, defect data is indispensable for project management. Large-scale projects may include several thousands of defects. These defects may be found in each stage of the project by many people. During the process, it often happens that the person that identifies and reports a defect is not the same person who corrects the defects [7].

The information on defects includes the number of defects that are identified through various defect detection activities. Therefore, all defects that are discovered in the requirement examination, design examination, code examination, unit test, and other stage are recorded. The distribution data of the number of defects found in different stages of the project are used in creating the Process Capability Baseline [8].

2.2 Defect Removal Efficiency

The DRE(Defect Removal Efficiency) of the defect detection stage is the ratio of the number of defects discovered at a stage against the total number of defects that appears when the stage is being processed. The more effective the DRE, the lower the possibility of undetected defects[3][11]. This shows that increasing the DRE (Defect Removal Efficiency) is a method to improve productivity. DRE is calculated as follows.

DRE = E(E+D), E= Number of defect found at relevant S/W development step(e.g : Number of defect found at request analysis step), D= Number of defect found at next S/W development step (e.g : Defect number that defect found at design step is responsible for defect of request analysis step) Ideal value of DRE is 1 or 0(no error), and this displays that no defect on the project.

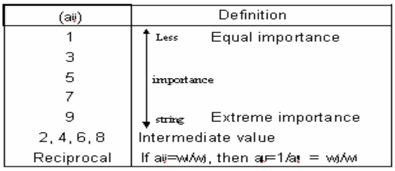
2.3 Analytic Hierarchy Process

The Analytic Hierarchy Process is a method for formalizing decision making where there are a limited number of choices but each has a number of attributes and it is difficult to formalize some of those attributes. Note in this example, we did not collect any data (like miles from a preferred point or salary numbers). Instead, we use phrases like ``much more important than" to extract the decision makers preferences[9].

The AHP has been used in a large number of applications to provide some structure on a decision making process. Note that the system is somewhat ad-hoc (why 1-9 range?) and there are a number of “hidden assumptions” (if i is weakly preferred to j and j weakly preferred to k, then a consistent decision maker must have i absolutely preferred to k, which is not what my idea of the words means). A matrix was used to compare the defect causes. The matrix shows the relation by listing each importance from 1 to 9 [9].

The measurement standard table of the Comparison Matrix is shown in table1.

Table 1. Comparisons Matrix measurement standard table



(a_{ij})	Definition
1	Less Equal importance
3	
5	importance
7	
9	string Extreme importance
2, 4, 6, 8	Intermediate value
Reciprocal	If $a_{ij}=w_i/w_j$, then $a_{ji}=1/a_{ij} = w_j/w_i$

The importance among the items of each stage were quantified based on table 2. This time, the differences of the item grade of each stage were compared. For example, if an item is Extra High(EH) and the other item is N, the importance between the two items is triple the difference. This is due to two grade differences. The same relation applies to table 2.

Table 2. A ratio production table of Grade vs importance difference

Grade difference between item	Grade vs importance difference ratio
1	Double
2	Triple
3	Quadruple
4	Quintuple
5	Sextuple

2.4 COQUALMO

In software estimation, it is important to recognize the strong relationships between Costs, Schedule and Quality. They form three sides of the same triangle. Beyond a certain point (the "Quality is Free" point), it is difficult to increase quality without increasing either the cost or schedule or both for the software under development. Similarly, development schedule cannot be drastically compressed without degrading the quality of the software product and/or increasing the cost of development. Software estimation models can play an important role in facilitating the balancing of these three factors[15].

3 Theory and Case Study

This chapter will develop a questionnaire to identify defects of the schedule management that occur during actual projects of companies. Therefore, the structure and contents of the questionnaire to detect the defect and analyze the result are presented in this chapter. Furthermore a defect management opportunity tree based on the analyzed result is provided. As for the domain of the project, digital systems were selected and 21 projects of four companies were chosen as target companies [12][13] [14].

3.1 Concept of Defect Management Opportunity Tree

We would like to be able to estimate how a software system's defect content change during its development, as well as to determine the locations with the highest concentration of faults. Prior to testing there is often little information regarding the number and location of errors. Surrogates based on changes in software structure can be used to estimate the rates of defect insertion into a system. These rates can be used to monitor the number of defects inserted into a system and the residual defect content, and to identify portions of a system requiring additional fault detection and removal resources. In this paper describes a surrogate developed, the measurement system implemented for taking the required measurements, and current efforts to extend this work.

We are provide opportunity tree algorithm next following.

1. Select goal(decision the business goal)
2. Select subgoal
3. Make the questionnaire for reliable verification
4. Decision goal and subgoal by Analytic Hierarchy Process
5. Provide solution for goal and subgoal
6. Design the opportunity tree

3.2 Requirement Analysis

A requirement detection questionnaire was made to identify defects of the schedule management opportunity tree in the ongoing project of a company.

Another purpose is to discover defects and analyze the causes between each defect. Questions that must be examined in the questionnaire were divided as below by classifying defects that exist in each defect issues. The structure of the questionnaire for each defect issue and frequency is as follows.

Table 3. Schedule management OT items

	Explain
Choose the lifecycle for the schedule management	Define the lifecycle for the schedule management
Identify the project planning and risk items	Analysis of the project plan and prevention of the risk items
Investigation of the rationale for the prevention of project delay	Investigation of the rationale for the prevention of project delay
Construction of organization for the schedule management	Construction of enterprise organization for the schedule management

Table 4. Frequency of Defect management OT items

	Frequency	Percent
Choose the lifecycle for the schedule management	50`	33
Identify the project planning and risk items	56	36.9
Investigation of the rationale for the prevention of project delay	36	23.5
Construction of organization for the schedule management	10	6.6
Total	152	100.0

To decide OT items of subgoal, 13 detailed categories were made for the questionnaire at the requirement analysis phase. Detailed categories for defect detection at the requirement analysis stage reflect customers' requirements and confirm how the requirements are satisfied. The requirements are classified in order to confirm consistency. Each detailed category and frequency is shown in table 5.

Table 5. Frequency of detailed subgoal of items

Items	Frequency	Percent
Create the lifecycle that the suitable project are tailered	22	9.2
Measurement Method for the progress management (Metric, Review of the Technology and management)	18	7.6
Building the development environment	15	6.3
Check the validity of the technology (Research, Simulation, Prototyping)	24	10.1
Decision the product of release time	12	5.1
Define of the objectives and needs, constraint for the system	24	10.1
Identify the relationship between processes and articles	22	9.2
Economy (Return On Investment, Market)	21	8.8
Select the lifecycle by the scale and complexity of project	25	10.5
Setting the Project schedule by the estimation of work breakdown structure and infrastructure	17	7.1
Validity of legal	15	6.3
Identify the project plan to the developer	11	4.6
Assessment the quality assurance	12	5.0
Total	188	100.0

3.2.1 Select the Lifecycle for the Schedule Management

The structure of the select the lifecycle for the schedule management questionnaire in the OT is shown in table 6. The purpose of OT a detailed category is to confirm that schedule lifecycle in the project development.

Table 6. Rating of the select the lifecycle for the schedule management of items

Items	Rating
1. Create the lifecycle that the suitable project are tailered	Nominal
2. Measurement Method for the progress management (Metric, Review of the Technology and management)	Low
3. Decision the product of release time	Very Low
4. Setting the Project schedule by the estimation of work breakdown structure and infrastructure	High

Defects in respective stages are presented in AHP. Therefore this chapter will analyze and measure the identified defect causes. To this end, the hierarchical structure for decision-making has been schematized, and the weight of the defect item in each stage has been graded into six dimensions. The graded items were analyzed based on a comparison metrics [10], and the geometric means of the schedule defect items were calculated in order to identify its relation with the causes.

The decision-making structure and graded items in each stage are as below.

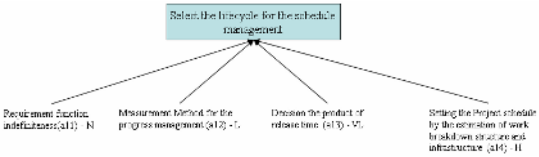

Select the lifecycle for the schedule management
Requirement function indefiniteness(a11) - N
Measurement Method for the progress management (a12) - L
Decision the product of release time (a13) - VL
Setting the Project schedule by the estimation of work breakdown structure and infrastructure (a14) - H

Fig. 1. Cause rating scale of schedule defect data collection and identify

The defect cause rating scale structure is divided into each of the defect OT items and each stage has been further categorized in detail. The rating scale is divided into six dimensions: Very Low(VL), Low(L), Normal(N), High(H), Very High (VH), Extra High (EH). This means that, among the six dimensions, priority increases as it nears EH, and the priority decreases when as it approaches VL.

In the case of Fig. 1, at the select the lifecycle for the schedule management item, a rating scale was made based on the weight of each item analyzed in AHP. The result showed that the "Setting the Project schedule by the estimation of work breakdown structure and infrastructure" item had the most defects.

In table 7, the importance was compared with other items based on a11, a12, a13, a14 columns. For example, according to the analysis, between a11 and a12, a11's defect item is Nominal and a12 is Low. Therefore, the difference of grade between items is 1 by table 7. Therefore, the difference between the grade and importance is double. The Compare metrics of the select the lifecycle for the schedule management are as below.

Table 7. Compare metrics of the select the lifecycle for the schedule management

	a11	a12	a13	a14
a11	1	2	2	1/2
a12	1/2	1	1	1/3
a13	1/2	1	1	1/3
a14	2	3	3	1

One of the purposes here is to calculate the importance of defect data collection and identify item based on the comparison metrics from table 7. The greater the importance of each item, the higher the probability of causing a defect. This was calculated by geometric means.

A production table of each stage is as follows.

Table 8. Calculation of OT item importance for the select the lifecycle for the schedule management

Item	Result value
a11	$(1 \text{ x } 2 \text{ x } 2 \text{ x } 1/2)^{1/4} = (1/3)^{1/4} = 1.19$
a12	$(1/2 \text{ x } 1 \text{ x } 1 \text{ x } 1/3)^{1/4} = 3^{1/4} = 0.64$
a13	$(1/2 \text{ x } 1 \text{ x } 1 \text{ x } 1/3)^{1/4} = 3^{1/4} = 0.64$
a14	$(2 \text{ x } 3 \text{ x } 3 \text{ x } 1)^{1/4} = (1/3)^{1/4} = 1.68$

Items of defect data collection and identify was produced by a geometric average.

The result of the analysis showed that, al4 and was the select the lifecycle for the schedule management causes.

Table 9. Solution of select the lifecycle for the schedule management

Items	Solution
1. Create the lifecycle that the suitable project are tailered	1. Identify of the project objectives 2. Analysis of the lifecycle by the project 3. Analysis of the requirement
2. Measurement Method for the progress management (Metric, Review of the Technology and management)	1. Analysis of the job 2. Making of the technical Analysis document
3. Decision the product of release time	1. Define the milestone of the development stages 2. Define the checklist for the release time
4. Setting the Project schedule by the estimation of work breakdown structure and infrastructure	1. Analysis for the detail job 2. Analysis of the relationship between jobs

We are provide the solution about select the lifecycle for the schedule management. The solution is like the table 9.

3.2.2 Identify the Project Planning and Risk Items

The structure of the Identify the Project planning and risk items questionnaire in the OT is shown in table 10. The purpose of OT a detailed category is to confirm that schedule lifecycle in the project development.

Table 10. Rating of the Identify the Project planning and risk items

Items	Rating
1. Building the development environment	Low
2. Define of the objectives and needs, constraint for the system	Very High
3. Identify the relationship between processes and articles	Very High
4. Setting the Project schedule by the estimation of work breakdown structure and infrastructure	Nominal
5. Identify the project plan to the developer	Low

Defects in respective stages are presented in AHP. Therefore this chapter will analyze and measure the identified defect causes. To this end, the hierarchical structure for decision-making has been schematized, and the weight of the defect item in each stage has been graded into six dimensions. The graded items were analyzed based on a comparison metrics [10], and the geometric means of the schedule defect items were calculated in order to identify its relation with the causes.

The decision-making structure and graded items in each stage are as below.

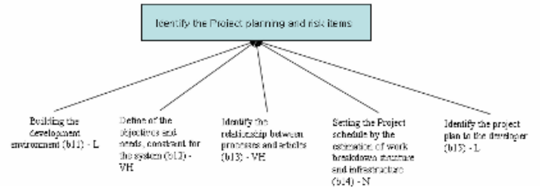

Identify the Project planning and risk items
Building the development environment (b11) - L
Define of the objectives and needs, constraint for the system (b12) - VH
Identify the relationship between processes and articles (b13) - VH
Setting the Project schedule by the estimation of work breakdown structure and infrastructure (b14) - N
Identify the project plan to the developer (b15) - L

Fig. 2. Cause rating scale of the Identify the Project planning and risk items

The defect cause rating scale structure is divided into each of the OT items and each stage has been further categorized in detail. The rating scale is divided into six dimensions: Very Low(VL), Low(L), Normal(N), High(H), Very High (VH), Extra

High (EH). This means that, among the six dimensions, priority increases as it nears EH, and the priority decreases when as it approaches VL.

In the case of Fig.2, at the select the Identify the Project planning and risk items, a rating scale was made based on the weight of each item analyzed in AHP. The result showed that the "Define of the objectives and needs, constraint for the system" and "Identify the relationship between processes and articles " item had the most defects.

In table 11, the importance was compared with other items based on b11, b12, b13, b14 columns. For example, according to the analysis, between b11 and b12, a11's defect item is Low and a12 is Very High. Therefore, the difference of grade between items is 4 by table 11. Therefore, the difference between the grade and importance is Quintuple. The Compare metrics of the select the lifecycle for the schedule management are as below.

Table 11. Compare metrics of the Identify the Project planning and risk items

	b11	b12	b13	b14	b15
b1	1	1/5	1/5	1/3	1
b12	5	1	1	3	5
b13	5	1	1	3	5
b14	3	1/3	1/3	1	3
b15	1	1/5	1/5	1/3	1

One of the purposes here is to calculate the importance of defect data collection and identify item based on the comparison metrics from table 11. The greater the importance of each item, the higher the probability of causing a defect. This was calculated by geometric means.

A production table of each stage is as follows.

Table 12. Calculation of OT item importance for the Identify the Project planning and risk items

Item	Result value
b11	(1 x 1/5 x 1/5 x 1/3 x 1)$^{1/5}$ = (1/75)$^{1/5}$ = 0.421
b12	(5 x 1 x 1 x 3 x 5)$^{1/5}$ = 3$^{1/5}$ = 2.37
b13	(5 x 1 x 1 x 3 x 5)$^{1/5}$ = 3$^{1/5}$ = 2.37
b14	(3 x 1/3 x 1/3 x 1 x 3)$^{1/5}$ = (1/3)$^{1/5}$ = 1
b15	(1 x 1/5 x 1/5 x 1/3 x 1)$^{1/5}$ = (1/75)$^{1/5}$ = 0.421

Items of defect data collection and identify was produced by a geometric average.

Table 13. Solution of the Identify the Project planning and risk items

Defect items	Solution
1. Building the development environment	1. Analysis of the functional and nonfunctional requirement 2. Provide the human resource for project 3. Building the infrastructure for project
2. Define of the objectives and needs, constraint for the system	1. Analysis of the project objectives 2. Analysis of the project risk items
3. Identify the relationship between processes and articles	1. Identify the output for the each of the processes 2. Analysis order of the job 3. Define the priority of the job
4. Setting the Project schedule by the estimation of work breakdown structure and infrastructure	1. Define the rationale for the project schedule 2. Define the infrastructure for the project schedule
5. Identify the project plan to the developer	1. Repeatable Confirm of the project plan 2. Check the progress of the each of schedule unit

The result of the analysis showed that, bl2, b13 and was the Identify the Project planning and risk items causes.

We are provide the solution about select the lifecycle for the schedule management. The solution is like the table 13.

4 Introduction to Effectiveness Analysis of Defect Management Opportunity Tree

We analyzed aspect that defect manage is achieved efficiently to analyze effectiveness of defect management opportunity tree. Also, we present effectiveness of introduction through if productivity improves because defect of whole project is reduced through defect management opportunity tree.

4.1 Defect Removal Efficiency Analysis

Purpose of defect management opportunity tree design improves quality of product and heighten productivity. Therefore, when we applied defect management opportunity tree in actuality project, we wish to apply defect exclusion efficiency (Defect Removal Efficiency). to measure ability of defect control activity.

After apply defect management opportunity tree, defect exclusion efficiency analysis [9] investigated defect number found at relevant S/W development step and defect number found at next time step in terms of request analysis, design and coding stage. Production of defect exclusion efficiency is as following. DRE = E/(E+D)

Ideal value of DRE is 1, and this displays that any defect does not happen to S/W.

Table 14. Table of defect removal efficiency

	Number(%) of defect found at relevant S/W development step (E)	Number(%) of defect found at next S/W development step (D)
Requirement	10	3
Design	15	3
Coding	5	1

Table14 is a table to inspect S/W development step defect number after Defect Trigger application.

0.769 = 10(10+3) (Requirement phase), 0.833 = 15/(15+3) (Design phase), 0.833 = 5/(5+1) (Coding phase)

If we save DRE at each S/W development step by Table21, it is as following. Therefore, because DRE is approximated to 1, when we remove defect by Defect Trigger, defect exclusion efficiency was analyzed high.

4.2.2 Calculation of COQUALMO

Defect Remove Effectiveness evaluation is compare number of real defect with number of defect estimation each of development phase by COQUALMO. Also, we analyze defect remove capability by Defect Remove Effectiveness method[8].

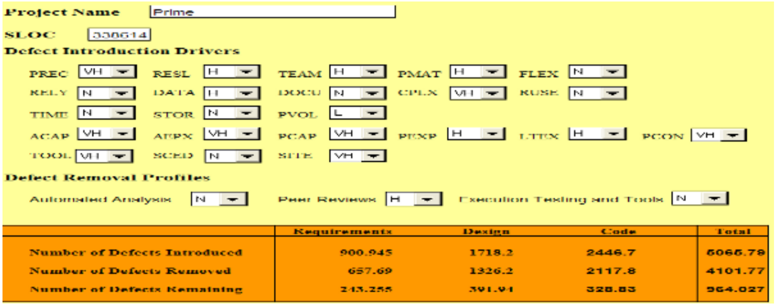
Real project defect is like the following. Table 3-2, 3-3 is apply to SPI (Software Process Improvement).

Table 15. Number of Defects Introduced

MBASE/Rational Model (Waterfall Model)	Inception (Requirements)	Elaboration (Product Des)	Construction (Development)	Transition	Total
No. of Requirements Defects	156	103	32	-	291
No. of Design Defects		346	211	-	557
No. of Code Defects			809	-	809
TOTAL	156	449	1052	-	1657

Table 16. Number of Defects Removed

MBASE/Rational Model (Waterfall Model)	Inception (Requirements)	Elaboration (Product Des)	Construction (Development)	Transition	Total
No. of Requirements Defects	156	103	32	-	291
No. of Design Defects		346	211	-	557
No. of Code Defects			809	-	809
TOTAL	156	449	1052	-	1657



Project Name Prime

SLOC 330614

Defect Introduction Drivers

PREC VH RESL H TEAM H PMAT H FLEX N
RELY N DATA H DOCU N CPLX VH RUSE N
TIME N STOR N PVOL L
ACAP VH AEPX VH PCAP VH PEXP H LTEX H PCON VH
TOOL VH SCED N SITE VH

Defect Removal Profiles

Automated Analysis N Peer Reviews H Execution Testing and Tools N

	Requirements	Design	Code	Total
Number of Defects Introduced	900.945	1718.2	2446.7	5065.78
Number of Defects Removed	657.69	1326.2	2117.8	4101.77
Number of Defects Remaining	243.255	391.94	328.83	964.027

Fig. 4. Calculation of Defect Number

There is 3 times differences between defect number of Table 3-2, 3-3 and defect number of COQUALMO. Also, such data must different 3~4 times to use of defect data in real project. Therefore, we can use of evaluation results.

Defect number estimation result is like the following.

1657(Real Project) X (ABOUT) 3.05 times = 5065(COQUALMO)

Defect number estimation is use of COQUALMO algorithm.

5 Conclusions

Chapter Three has presented the designing and analysis of a defect management opportunity tree based on defect causes.

In implementing a similar project, using such opportunity tree will help produce a more reliable result in terms of cost, quality, and schedule of the entire project by predicting the stages where defects occur and the contents of the defects.

In order to make this possible, further study is required on how this can be used to enhance the defect reduction estimates by using the opportunity tree information.

The direction of further study should be to develop a opportunity tree framework by providing detailed items of the defects, and to develop a system that is capable of predicting defects by inputting major items on the web.

References

1. Annual Research Review, USC/CSE workshop reports, oct. 2002
2. Barry W. Boehm, "SOFTWARE COST ESTIMAITION WITH COCOMO II", Prentice-Hall PTR, 2000
3. Software Process Improvement Forum, KASPA SPI-7, dec. 2002
4. George Eckes, The Six Sigma Revolution, John Willey & Sons, 2001
5. N. Fenton and N. Ohlsson " Quantitative analysis of faults and failures in a complex software system" , IEEE Trans. Software Eng., 26, 797-814, 2000.
6. McCall, P.K. Richards and G.F. Walters, Factors in software quality." Vol 1, 2 and 3. Springfield VA., NTIS, AD/A-049-014/015/055, 1997
7. Vouk, Mladen, A., " Software Reliability Engineering" , Tutorial Notes? Topics in Reliability & Maintainability & Statistics, 2000 Annual Reliability and Maintainability Symposium, Los Angeles, CA, 2000 January 24-27.
8. Wohlin, Runeson, " Defect Content Estimations from Review Data" , Proceedings International Conference on Software Engineering ICSE 400-409, 1998.
9. Roger S. Pressman "Software Engineering" Mcgraw-Hill International edition, 1997.
10. Actin Focused Assessment for Software Process Improvement, Tim Kasse, Artech House, 2002
11. Robert H. Dunn, " Software defect removal" , McGraw-hill, 1984.
12. Eun-ser Lee , Malrey Lee, Development System Security Process of ISO/IEC TR 15504 and Security Considerations for Software Process Improvement(LNCS), 2005.05, ICCSA 2005
13. Eun-ser Lee, Tai-hoon Kim, Introduction of Development Site Security Process of ISO/IEC TR 15504, Knowledge-Based Intelligent Information and Engineering Systems(LNCS), 2004.09
14. Eun-ser Lee , Kyung-Whan Lee, KeunLee, Development Design Defect Trigger for Software Process Improvement, Software Engineering Research and Applications (LNCS), 2003.06
15. V. Basili, G. Caldiera and D. Rombach, "The Experience Factory", Encyclopedia of Software Engineering" Wiley 1994.

CTL Model Checking for Boolean Program[*]

Taehoon Lee, Gihwon Kwon, and Hyuksoo Han

Department of Computer Science, Kyonggi University,
San 94-6, yiui-dong, Youngtong-Gu, Suwon-si, Kyonggi-do, Korea
{taehoon, khkwon}@kyonggi.ac.kr
College of Computer Software and Media Technology, Sangmyung University,
Hongj-dong Jongno-gu, Seoul, Korea
hshan@smu.ac.kr

Abstract. Nowadays, there are some subtle errors in a software system. So verification technique is very important. The one of important verification technique is model checking technique. Model checking is a technique to verify behavior of system with desired property. There are many researches about software model checking. As a result, predicate abstraction techniques are proposed and many tools for C or Java are developed. In general, there are two types of properties: The first is the safety properties. And other one is liveness properties. Most software model checking tools can only verify safety properties. In this paper, we describe CTL model checking algorithm based on Boolean program and describe model checking tool for Simple Java program which used in Lego robot to verify liveness property. Our model checking tool can check not only safety property but also liveness property and we describes case study verifying safety property and liveness property of LEGO robot.

1 Introduction

Model checking is a formal verification technique for verifying finite state systems.[1] Given finite state model M and temporal formula ϕ , model checking determines whether the model satisfies the formula, written $M \models \mathrm{AG}\neg\phi$. The most widely used verification techniques are simulation and testing. But these techniques can cover only a limited set of possible behaviors. So these techniques can prove "there is property violation." But they can't prove "there is no property violation." In here, property violation can be bug. Model checking differs from these traditional verification methods in several aspects. Most of all, Model Checking perform an exhaustive search of state space of system. Of course, it can prove "there is no propertiy violation." But Model checking technique is not treated as major software verification. Because of most software system has infinite state space. But model checking can handle only finite state space.

There are many software model checking tools such as SLAM [2], JavaPathFinder[3], Bandera[4], BLAST[5], MAGIC[6]. To Verity the source code of program,

[*] This work was supported by grant No.(R01-2005-000-11120-0) from the Basic Research Program of the Korea Science & Engineering Foundation.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1081 – 1089, 2006.

the technique that abstract infinite state space into finite state space is required. One of successful technique is Predicate abstraction [7] which used by SLAM. SLAM has three verification phases.

The first step of this algorithm is generating Boolean program from the C program and set of predicate E that is condition expression containing no function calls. Boolean program is guaranteed to be an abstraction of the C program in the following sense; any feasible execution path of the C program is a feasible execution path of Boolean program. Of course, there may be feasible path of Boolean program that are infeasible in the C program. Next step is to determine whether or not the label SLIC_ERROR is reachable in Boolean program. If the answer is no, C program can't reachable label SLIC_ERROR. If the answer is yes, the SLAM's model checking tool produce a path leading to the error state. In this step, there is a one question. Does path represent a feasible execution path of C program? The the SLAM's refinement tool takes a C program and an error path as an input it then uses verification condition generation to determine if the path is feasible. The answer may be "yes" or "no" If the answer is yes, then an error path has been found. If the answer is no then the SLAM's refinement tool uses a new algorithm to identify a small set of predicate that explain why path is infeasible.

In general, there are two types of properties to be verified. One is safety properties. And other one is Liveness properties. Safety property is the property that something bad will not happen. Liveness property is the property that something good will happen. Deadlock or error state represents something bad. Guarantee of termination or response of request is liveness property. In SLAM, property that can be verified is only safety property. We extend CTL model checking algorithm to verify Boolean program. Figure 1 shows our model checking tool's framework. This paper presents our tool to perform CTL model checking.

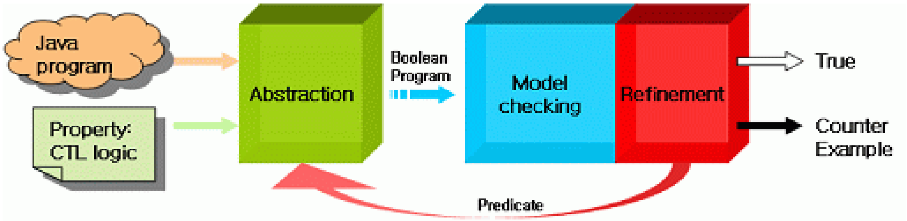


Fig. 1. Tool for Model Checking Java

The rest of the paper is organized as follows: firstly we describe general CTL model checking algorithm. After which we describe CTL model checking for Boolean program. After which we describe case study. Finally some conclusions are given in Section 4.

2 Preliminaries

This section gives a brief overview on CTL model checking.

2.1 Model

A model is 4-tuple. M=<S,I,R,L>, where S is a finite set of states, $I \subseteq S$ is the set of initial states, $R \subseteq S \times S$ is the transition relation, and the function $L:S \rightarrow 2^{AP}$ assigns to each state a set of atomic propositions that are true at that state, where AP is a finite set of atomic propositions. The transition relation R is assumed to be total: $\forall s \in S \cdot \exists s' \in S \cdot (s,s') \in R$ that is, for every state $s \in S$, there exist a successor $s' \in S$ with $(s,s') \in R$. a path is an infinite sequence of states in which each consecutive pair of states belongs to R.

2.2 Property

Desired properties about a model can be specified in the Computation Tree Logic. The syntax of CTL formula is defined as follows:

$$\phi, \psi ::= \text{true} \mid p \mid \neg\phi \mid \phi \vee \psi \mid \text{EX}\ \phi \mid \text{EF}\ \phi \mid \text{EG}\ \phi \mid \text{E}(\phi\ \text{U}\ \psi) \mid \text{AX}\phi \mid \text{AF}\ \phi \mid \text{AG}\phi \mid \text{A}(\phi\ \text{U}\ \psi)$$

As usual, E is the existential path quantifier, A is the universal path quantifier, X is the next-time operator, F is the future operator, G is global operator, and U is the *until* operator. Intuitively, EX ϕmeans that there is a successor state at which ϕ is true, EF ϕ means that there is a state at which ϕ is eventually true, EG ϕ means that ϕ is always true for some path, E(ϕ U ψ) means that for some path, ϕ remains true until ψ becomes eventually true. Assume a fixed model M. We write $s_0 \models \phi$ if the CTL formula ϕ is true at state s_0. The truth value of a CTL formula at state s_0 is defined as follows:

$s_0 \models \text{true}$

$s_0 \models p$ iff $p \in L(s_0)$

$s_0 \models \neg\phi$ iff $s_0 \not\models \phi$

$s_0 \models \phi \vee \psi$ iff $s_0 \models \phi$ or $s_0 \models \psi$

$s_0 \models \text{EX}\ \phi$ iff there exists $(s_0, s_1) \in R$ such that $s_1 \models \phi$

$s_0 \models \text{EF}\ \phi$ iff for some path $s_0, s_1, s_2, \ldots$, thers exists $i \geq 0$, $s_i \models \phi$

$s_0 \models \text{EG}\ \phi$ iff for some path $s_0, s_1, s_2, \ldots$, for all $i \geq 0$, $s_i \models \phi$

$s_0 \models \text{E}(\phi\ \text{U}\ \psi)$ iff for some path $s_0, s_1, s_2, \ldots$, there exists $0 \leq j < i$, for all j, $s_j \models \phi$ and $s_i \models \psi$

2.3 Model Checking

We sat that M satisfies ϕ, written $M \models \phi$, if $s \models \phi$ for each $s \in I$,ie, ϕ is true at every initial state of M. Given a model and a CTL formula, the CTL model checking determines whether the model satisfies the formula. For ant CTL formula ϕ, let $[\![\phi]\!]$ denote the set of states om which ϕ is true. Then the model checking algorithm is equivalent to determining whether the set of initial states is a subset of $[\![\phi]\!]$; i.e., the following equations hold;

$M \models \phi$ iff $I \subseteq [\![\phi]\!]$

$M \not\models \phi$ iff $I \not\subseteq [\![\phi]\!]$

For any state set X, we define $pre_{\exists}(X) = \{s \in S \mid \exists_{s' \in S} \bullet (s,s') \in R \wedge s' \in X\}$ as the set of states with a successor in X. the following equations hold for any CTL formulas:

$$
\begin{aligned}
\llbracket \text{true} \rrbracket &= S \\
\llbracket p \rrbracket &= \{s \mid p \in L(s)\} \\
\llbracket \neg\phi \rrbracket &= S \setminus \llbracket \phi \rrbracket \\
\llbracket \phi \vee \psi \rrbracket &= \llbracket \phi \rrbracket \cup \llbracket \psi \rrbracket \\
\llbracket \mathrm{EX}\, \phi \rrbracket &= pre_{\exists}(\llbracket \phi \rrbracket) \\
\llbracket \mathrm{EF}\, \phi \rrbracket &= \mu \mathrm{Z}.(\llbracket \phi \rrbracket \cup pre_{\exists}(\mathrm{Z})) \\
\llbracket \mathrm{EG}\, \phi \rrbracket &= \nu \mathrm{Z}.(\llbracket \phi \rrbracket \cap pre_{\exists}(\mathrm{Z})) \\
\llbracket \mathrm{E}(\phi\, \mathrm{U}\, \psi) \rrbracket &= \mu \mathrm{Z}.(\llbracket \psi \rrbracket \cup (\llbracket \phi \rrbracket \cap pre_{\exists}(\mathrm{Z})))
\end{aligned}
$$

where μ and ν are the least fixed-point and greatest fixed-point operators respectively. It can be shown that $pre_{\exists}$ and these fixed points can be computed in time linear in the size of the model. Because the formula is evaluated by computing predecessors of states, the CTL model checking algorithm is based on backward traversals.

3 CTL Model Checking for Boolean Program

Boolean Programs that are produced by predicate abstraction. Boolean program is a program which has Boolean variable. In this research, we aim to verify simple java program. The simple java program is based on JDK 1.2. Recursive call, multi thread and external library are not supported on simple java program. We make the tool which produce Boolean program from simple java program by predicate abstraction. Boolean program has grammar as follows

```
BP:= class*
Class:=var* method*
Var:= predicate STRING
Method:= STRING(VAR *) {  Statement *}
Statement := skip;
      | goto STRING;
      | STRING( expr+);
      | if(expr) then statement* else statement*
      | STRING = expr
      | assume( expr+ );
expr : =   | STRING( expr+);
      | STRING = expr
      | expr op expr
      | ( expr )
      | ! Expr
Op := "&" | "|"
```

Boolean program consists of *skip, goto, method call, assignment, if, assume* statements. The *Skip* statement moves program counter to the next program counter without changing value of variable. The *goto* statement moves program counter to the given program counter. The *Method call* statement moves program counter to first position of method. The *Assignment* statement changes the value of variable. The *If* statement moves program counter to then part when condition is true or moves program counter to else part when condition is false. If condition is not true or false, it can move to then part and else part. The *Assume* statement means that program assumes that state of program satisfy the condition of the *assume* statement. To model checking boolean program, we translate Boolean program to graph structure. The Flow graph is defined as follows

$$FG=(S, I, R)$$

S is a set of state. State s is $s = PC \times 2^{\gamma}, s \in S$ $I \in S$ is an initial state. $R : S \times S$ is a transition relation. *PC* means program counter of Boolean program. γ is mapping function that assigns a variable to a value. Example is as follows

$$V=\{x,y,z\}\ ,\ \gamma = \{(x,true),(y,false),\ (z,true)\}$$

Firstly, we define function which is needed to define state transition. *post(pc)* return next program counter. *post$_T$(pc)* return the program counter of then part in *if* statement. *post$_F$(pc)* return the program counter of else part in *if* statement. Σ is mapping function that map program counter to the value of *assume* statement. Δ_{cm} is mapping function that map variable name of caller to callee. *First$_M$(pc)* returns program counter of callee method. *ReturnM(pc)* returns program counter of caller method.

The initial state of Boolean program is a fist statement of main method. The Value of Variable is assigned to true. Each statement is used to add transition relation. We summarize it in Table 2.

Table 2. Transition Relation

statement	Transition relation
X=true	$((pc, \gamma),(post(pc), \gamma[x/true]))$
If	$((pc, \gamma),(post_T(pc), \gamma))$ $((pc, \gamma),(post_F(pc), \gamma))$
Assume	$((pc, \gamma),(post(pc), \gamma))$ where $\forall v \in V.\Sigma(pc)(v) = \gamma(v)$ $((pc, \gamma), (pc, \gamma))$ where $\exists v \in V.\Sigma(pc)(v) \neq \gamma(v)$
Method call	$((pc, \gamma),(First_m(pc), \gamma))$
Return	$((pc, \gamma),(Return_m(pc), \gamma))$
Skip	$((pc, \gamma),(post(pc), \gamma))$

If the current statement is variable assignment, current position of program is move to next position and the value of variable is changed to true or false or unknown. The *If* statement is move program counter to then-part and else-part. In case of The Assume statement, the *If* condition is satisfied, then program counter is move to next program counter. The *If* condition is not satisfied, then next program counter is set to current program counter. In the method call, if formal parameter exists, then Δ_{cm} is used to map caller's variable name to callee's variable name. if the *return* statement has variable, then Δ_{cm} is used to map callee's variable name to caller's variable name.

For example, given simple Boolean program,

```
      class a {
1.    public void main(  ) {

      Predicate C1: m <= 10;
      Predicate C2: x == 0;
      Predicate C3: r == true;

2.      C1 = true;
3.     C2 = true;
4.     C3 = false;

5.     L1:
6.     if( * ) {
7.            assume( !(C3));
8.            if( * ) {
9.                   assume(  C1);
10.                   C1 =!C1;
              }else {
11.                   assume( !(C1));
12.                   C2 = false;
13.                   C3 = true;
              }
14.           goto L1;
       }
       }
      }
```

We make following transition relation.

```
(1, (true,c2,c3) ) , ( 2,(true,c2,c3) )
(1, (false,c2,c3) ) , ( 2,(true,c2,c3) )
(2,(c1,true,c3) ) , ( 3,(c1,true,c3) )
(2,(c1,false,c3) ) , ( 3,(c1,true,c3) )
(3,(c1,c2,true) ) , ( 4,(c1,c2,false) )
(3,(c1,c2,false) ) , ( 4,(c1,c2,false) )
(4,(c1,c2,c3) ) , ( 5,(c1,c2,c3) )
(5,(c1,c2,c3) ) , ( 6,(c1,c2,c3) )
(6,(c1,c2,c3) ) , ( 7,(c1,c2,c3) )
(6,(c1,c2,c3) ) , ( 15,(c1,c2,c3) )
(7,(c1,c2,c3) ) , ( 8,(c1,c2,c3) )
(8, (c1,c2,c3) )  , ( 9,(c1,c2,c3) )
(8, (c1,c2,c3) )  , ( 11,(c1,c2,c3) )
(9, (false,c2,c3) )  , ( 9,(false,c2,c3) )
(9, (true,c2,c3) )  , ( 10,(true,c2,c3) )
(10, (true,c2,c3) )  , ( 14,(false,c2,c3) )
(10, (false,c2,c3) )  , ( 14,(true,c2,c3) )
(11, (false,c2,c3) ) , (12, (false,c2,c3) )
(11, (true,c2,c3) ) , (11, (true,c2,c3) )
(12, (c1, true,c3) )  , (13, (c1, false,c3) )
(12, (c1, false,c3) )  , (13, (c1, false,c3) )
(13, (c1, c2, true) ) , (14, (c1,c2,true) )
(13, (c1, c2, false) ) , (14, (c1,c2,true) )
(14, ( c1,c2,c3)  ) , ( 6,( c1,c2,c3) )
(15, ( c1,c2,c3)  ) , ( 16,( c1,c2,c3) )
```

In this example, each program statement corresponds to state transition. State transition consists of PC which defines program position and value of variable. When statement is the assignment statement *C1=true*, PC is moved to next PC and value of C1 is assigned to true. When current statement is the *if* statement, PC is moved to $post_T(pc)$ or . $post_E(pc)$. When statement is the *assume* statement, we assume that condition of the *assume* statement is true. So if condition is true, PC is moved to next PC. If condition is false, then PC is not moved to next PC.

To perform CTL model checking, we redefine $pre_\exists$ operator.

$$pre_\exists(i,j) = \{(x,y) \mid (x,i) \in PC \times PC \wedge (y,i) \in 2^\gamma \times 2^\gamma \wedge ((x,y),(i,j)) \in R\}$$

Using $pre_\exists$ operator, we redefine CTL model checking algorithm as follows

$$[\![(pc,\gamma)]\!] = \{(pc,\gamma') \mid \gamma' = \gamma \wedge pc \in PC\}$$
$$[\![\neg(pc,\gamma)]\!] = \{(pc,\gamma') \mid pc' \in PC \setminus pc \wedge \gamma' \in 2^\gamma\} \cup \{(pc,\gamma') \mid \forall a \in V.\gamma'[a] = \neg\gamma[a]\}$$
$$[\![EX(pc,\gamma)]\!] = pre_\exists(pc,\gamma)$$
$$[\![EF(pc,\gamma)]\!] = \mu Z.([\![(pc,\gamma)]\!] \cup pre_\exists(Z))$$
$$[\![EG(pc,\gamma)]\!] = \nu Z.([\![(pc,\gamma)]\!] \cap pre_\exists(Z))$$
$$[\![AG(pc,\gamma)]\!] = [\![\neg EF\neg(pc,\gamma)]\!]$$
$$[\![AF(pc,\gamma)]\!] = [\![\neg EG\neg(pc,\gamma)]\!]$$
$$[\![AX(pc,\gamma)]\!] = [\![\neg EX\neg(pc,\gamma)]\!]$$

Each state is represented as (*pc*, *γ*). *pc* represents current program counter. *γ* repressentes value of variable in current *pc*. The negation of the state is divided into two parts. In all *pc* without current *pc*, the value of variable can assign as true or false. In current pc, the value of variable is the negation of the current value. The μ,ν which used in EF and EG represent least fixed point and greatest fixed point. We compute state which satisfies CTL formula using redefined CTL algorithm. Given CTL formula *φ*, if initial state I$\in [\![\varphi]\!]$, then Boolean program satisfy CTL formula. If initial state I$\notin [\![\varphi]\!]$, then Boolean program doesn't satisfy CTL formula.

In our work, we perform predicate abstraction in order to reduce state space. Predicate abstraction produce Boolean program which has more behavior than original program. Boolean program is over-approximation of simple java program. Because of Boolean program has more behavior, all CTL operator can't be applied in our approach. ACTL operator can be applied in our approach.

4 Case Study

We make the model checking tool which base on CTL model checking for Boolean program. This tool is aimed to LEJOS[11]. LEJOS is subset of Java. It can make a program for a Lego robot with Java. Currently, we implement abstract and model checking tool. And refinement tool is implementing. As an example, we implement the extended line tracking robot. It traverses the maze based on left hand rule. If signal from external environment is generated, then the robot moves by the signal. The signal is generated by computer and communicated by IR-sensor. Simple overview is in figure 1,

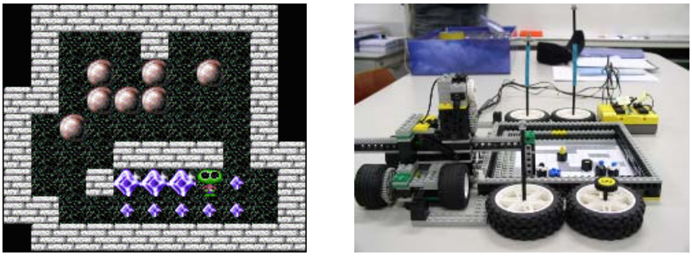


Fig. 1. Pushpush 50th level, it's robot simulation system

In this example, we are identified the critical properties. One of that is "always, game is not in end state". Another of that is "always, if sensing occurs, then lr_cycle_cnt is increase on all paths" CTL formulas of this properties is as follows.

AG not (*game*== "Clear")
AG (*sensing* ==true => AF (*lr_cycle_cnt*++))

The first property is safety property. It means that robot system can not move the ball to the goal position. We insert statement game= "Clear" into program. By our model checking tool, if program can reach that statement, then we can know the path to the statement by counterexample.

The second property is liveness property. The statement *sensing==true* means that sensing is occurred. Robot arm move one cell in game system. And the statement *lr_cycle_cnt*++ means that robot arm prepare to move next cell. Second CTL formula means that when sensing occurs, robot prepare to move next cell. It is not be verified in traditional model checking tools like SLAM or BLAST.

We compare general purpose model checking tool NuSMV and our model checking tool. Firstly, we perform predicate abstraction on Robot software. And translate NuSMV's input language. We can verify two properties. NuSMV verify the properties within 1.2 second and 12Mb. But our model checking tool can verify the same properties in 0.5second time and 6 Mb.

5 Conclusions

Software Model checking is very popular research topic. In many software model checking tools, verifiable property is restricted to safety property. To verify temporal property, we propose CTL model checking for Boolean program and make tool for LEJOS. In our case study, we describe how to verify liveness property by CTL model checking algorithm. Our model checking tool has better performance then NuSMV.

Our tool which we make is accepting restricted grammars. No thread support, no dynamic object creation, no multiple array. Currently, we plan to extend our tool to verify general java program. We plan to apply our tool to various java systems.

References

1. E.M. Clarke, O.Guumber and D.A. Peled. Model Checking, The MIT Press, 1999.
2. E.M. Clarke, O.Guumber, S. Jha, Y. Lu and H. Veith, "Progress on the State Explosion Problem in Model Checking", LNCS 2000, pp.154-169, 2000
3. S.Graf and H. saidi "Construction of Abstraction State Graphs with PVS", in Proceedings of Computer Aided Verification, pp72-83, 1997.
4. T.Ball, R. Majumdar, T. Millstein and S.K. Rajamani, "Automatic Predicate Abstraction of C programs", SIGPLAN Notices, Vol 36, No5, pp.203-213,2001
5. T.A. Henzinger , R. Jhala, R. Majumdar and G. sutre, "Lazy Abstraction", in Proceeding of Principles of Programming Languages, pp58-70, 2002.
6. S. Charki, E.M. Clarke, A. Groce, S. Jha and H. Veith, "Modular Verification of software Components in C", IEEE Transactions on Software Engineering, Vol30, No.6, pp388-402,2004.
7. J. Corbett, et al, "Bandera: Extracting Finite-state Models from Java Source Code", in proceedings of Internal Conference Software Engineering, 2000.
8. E. A. Emerson, "Temporal and modal logic",in the Handbook of Theoretical Computer Science: Formal Models and Semantics, Elsevier, pp.955-1072, 1990.
9. G. Farrari, A. Gombos, S. Hilmer, J. Stuber, "Programming Lego Mindstorms with Java: The Ultimate Tool for Mindstorms Maniacs", Syngress, April 2002.
10. C. Eisner ," Model Checking the garbage collexction mechanism of SMV", Electronic Notes in Theoretical computer Science Vol. 55, Elsevier Science Publishers, 2001

Grid Service Implementation of Aerosol Optical Thickness Retrieval over Land from MODIS

Yincui Hu[1], Yong Xue[1,2,*], Guoyin Cai[1], Chaolin Wu[1], Jianping Guo[1,3], Ying Luo[1,3], Wei Wan[1,3], and Lei Zheng[1,3]

[1] State Key Laboratory of Remote Sensing Science,
Jointly Sponsored by the Institute of Remote Sensing Applications of
Chinese Academy of Sciences and Beijing Normal University,
Institute of Remote Sensing Applications, Chinese Academy of Sciences,
P. O. Box 9718, Beijing 100101, China
[2] Department of Computing, London Metropolitan University,
166-220 Holloway Road, London N7 8DB, UK
[3] Graduate School of the Chinese Academy of Sciences, Beijing, China
huyincui@163.com, y.xue@londonmet.ac.uk

Abstract. To derive the actual land surface information quantitatively, the atmospheric effects should be correctly removed. Atmospheric effects dependent on aerosol particles, clouds and other atmosphere conditions. Aerosol parameters can be retrieved from the remotely sensed data. The retrieved aerosol characters can also be applied to environmental monitoring. To retrieval the aerosol optical thickness over land, many methods have been developed. The most popular one is the dark dense vegetation method. But it is confined to vegetation fields. The SYNTAM method can be used to retrieval aerosol optical thickness over land from MODIS data, no matter whether the land is dark or bright. In this paper, the SYNTAM method is applied to MODIS data for the retrieval of aerosol optical thickness over China. The retrieval process is complicated. And the EMS memory required is too large for a personal computing to run successfully. To solve this problem, the Grid environment is used. Our experiments were performed on the High-Throughput Spatial Information Processing Prototype System based on Grid platform in Institute of Remote Sensing Applications, Chinese Academy of Sciences. The aerosol optical thickness retrieval process is described in this paper. And the detail data query, data pre-processing, job monitoring and post-processing is discussed. Moreover, test results are also reported in this paper.

1 Introduction

Aerosol particles in the atmospheric play a complex role in optical remote sensing. Their absorption and diffusion characteristics alter the radiation reaching the sensor. Atmospheric aerosols affect global energy balance too. There are some in situ stations

[*] Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1090 – 1097, 2006.

to monitor atmospheric aerosols parameters, such as AERONET. But the stations are sparsely distributed. We can hardly gain the global distribution without the help of the remote sensing.

Aerosol parameters can be retrieved from the remotely sensed data. Retrieval of the aerosol optical thickness over sea from remote sensing data has been matured and routinely. But retrieval of the aerosol optical thickness over land remains difficult. Dark pixel method has been proposed among researchers to retrieve aerosol properties over dark pixels, such as water bodies and vegetation areas (Liu *et. al.*, 1996). Moderate-Resolution Imaging Spectroradiometer (MODIS) is one of instruments aboard NASA's Terra and Aqua satellites. MODIS acquire data in 36 spectral bands with spatial resolution 250m, 500m and 1km. When the temporal differences between the two satellite runs across the same region can be ignored, the Synergy of TERRA and AQUA MODIS data (SYNTAM) algorithm is used to retrieval aerosol optical thickness (Tang *et al.* 2005).

In the early days of retrieval, the MODIS data and programs have to be downloaded via Internet/Intranet or copied using mediums. This approach is labour intensive and requires human interactions to minimize the errors. Hailed as the next revolution after the Internet and the Web, Grid computing aggregates heterogeneous resources and provides hardware and software services, supporting application and services composition, workflow expression, scheduling, and execution management and service level agreements based allocation of resources. It has been an enabled environment for data sharing and processing.

Researchers and corporations have developed different types of grid computing platforms to support resource pooling or sharing. SETI@Home, Condor, and Alchemi harness idle CPU cycles from desktop computers in the network. Globus, EU DataGrid, and Gridbus allow sharing of computational and distributed data resources. The Grid applications in remote sensing have been studied by many researchers. The Grid architecture for remote sensing data processing is proposed by Aloisio *et al.* (2004) and the Grid platform of remote sensing data processing is developed (Aloisio *et al.* 2003).

Our research group has developed a Grid-based remote sensing environment, which is the High-Throughput Spatial Information Processing Prototype System in the Institute of Remote Sensing Applications, Chinese Academy of Sciences (Cai *et al.* 2004, Wang *et al.* 2004). In our Grid environment, the end users submit their application requirements to the Grid resource broker which then discovers suitable resources by querying the information services, schedules the application jobs for execution on these resources and then monitors their processing until they are completed. A more complex scenario would involve more requirements and therefore, Grid environments involve services such as security, information, directory, resource allocation, application development, execution management, resource aggregation, and scheduling. Software tools and services providing these capabilities to link computing capability and data sources in order to support distributed analysis and collaboration are collectively known as Grid middleware.

This paper focuses on the design and implementation of the services of AOT retrieval. First, we discussed the algorithm of AOT retrieval in Section 2. The architecture of the AOT retrieval services on Grid is introduced in Section 3. Finally, the implementation of services is provided in Section 4 and the future work is described.

2 SYNTAM Algorithm of AOT Retrieval

Moderate-Resolution Imaging Spectroradiometer (MODIS), aboard both NASA's Terra and Aqua satellites, acquire data in 36 spectral bands with spatial resolution 250m, 500m and 1km. The Synergy of Terra and Aqua MODIS data (SYNTAM) algorithm is used to retrieval aerosol optical thickness in this paper. The aerosol retrieval model bases on Eq. (1).

$$A_{j,\lambda_i} = \frac{(A_{j,\lambda_i}{}' b - a_j) + a_j(1 - A_{j,\lambda_i}{}')e^{(a_j - b)\varepsilon(0.00879\lambda_i^{-4.09} + \beta_j\lambda_i^{-\alpha})\sec\theta_j{}'}}{(A_{j,\lambda_i}{}' b - a_j) + b(1 - A_{j,\lambda_i}{}')e^{(a_j - b)\varepsilon(0.00879\lambda_i^{-4.09} + \beta_j\lambda_i^{-\alpha})\sec\theta_j{}'}} . \tag{1}$$

where j=1,2, respectively stand for the observation of TERRA-MOIDS and AQUA-MODIS; i=1,2,3, respectively stand for three visible spectral bands of central wavelength of 0.47μm, 0.55μm, 0.66μm; λ is the central wavelength. A is the Earth's surface reflectance. A' is the Earth's system reflectance (Tang, *et al.*2005). The process of AOT retrieval is shown in Figure1.

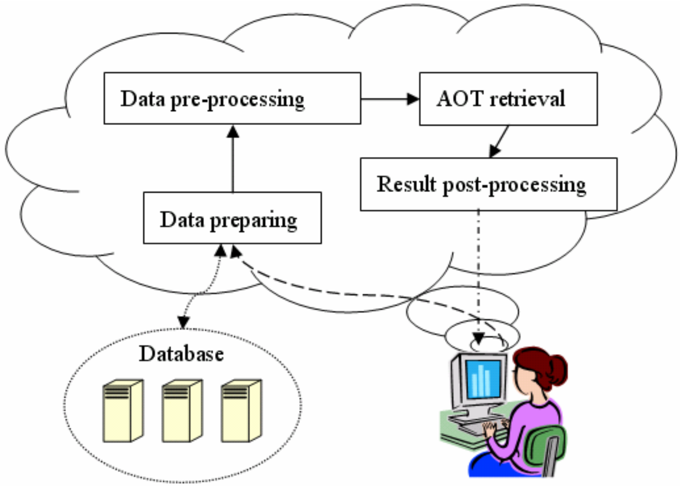


Fig. 1. The process of AOT retrieval

There are three basic components to retrieval AOT values. After the end users input their requirements, the data that meet the demand of the requirements must be found and pre-processed, such as calibration, cloud mask, geo-reference. Then the processed data are sent to the next step to retrieval AOT values. The retrieval results are post-processed (such as format transform) and then provided to the end users.

3 Implementation of AOT Retrieval Services on the Grid

The SYNTAM method is applied to retrieval aerosol optical thickness. The retrieval process is complicated. And the EMS memory required is too large for a personal computing to run successfully. To solve this problem, the Grid environment is used.

The retrieval process consists of data preparing, data pre-processing, SYNTAM computing and results post-processing. So the Grid service of the SYNTAM AOT retrieval is partitioned into four sub services correspondingly. These sub services are implemented orderly. When receive the user's order of the AOT retrieval service via grid portal, the Grid manager initialised an AOT retrieval service and run the data searching sub service to find the MODIS data among the data resource in the Grid pool. Then the data query results are sent to the data pre-processing service. The pre-process service sends the pre-process job to the computing resource and then collect the returned results. The pre-processed MODIS data finally transport to the SYNTAM processing service. After processing among computing resource in the Grid, the Grid collects all of the retrieved AOT results and then post-processes them. The final results are sent to the user via the Grid portal. The job status is monitored by a Grid pool manager. The architecture of the Grid service of the SYNTAM AOT retrieval is shown in Figure 2.

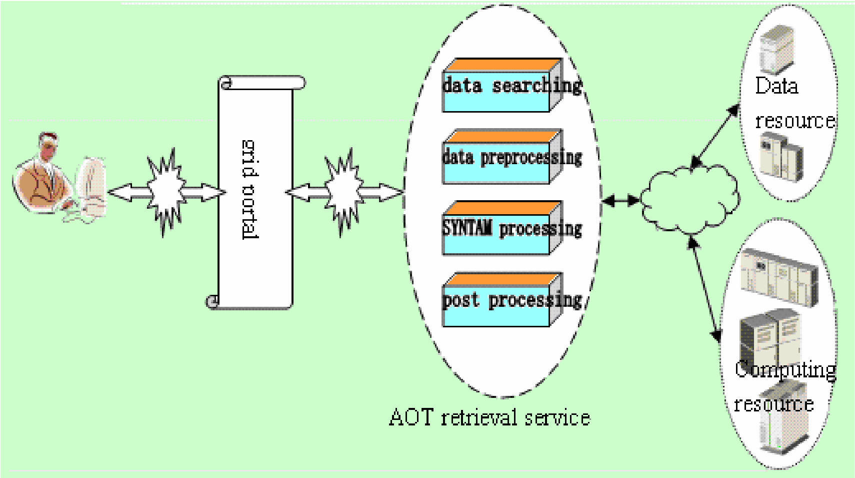


Fig. 2. Architecture of the grid service of the SYNTAM AOT retrieval

When the AOT retrieval program and the required data are store on the user's PC, the process is simplified by not considering the data preparing. When come to Grid environment, it is complicated because we must find the required data in the Grid pool and convert them to the right format that the program permits. Moreover, because we use the Grid resource to resolve the EMS memory problem, the job partition strategy and the results collection must be considered. We will discuss the data searching methods, data pre-processing, job management and post-processing of the collected results in the following paragraphs.

3.1 Data Query

The MODIS data resource is comprised of a set of Grid data nodes that are distributed in the Grid environment. The query strategy depends on the organization of the database. The MODIS database is usually file based and the files are stored in HDF format. The data resource and their metadata register to the Grid by registration services. The metadata describe the information of the MODIS data, which includes range, producer, quality, date and time, processing methods, satellite, and so on. To retrieve AOT by SYNTAM algorithm, the TERRA and AQUA MODIS data of the same region where the user chose are required. The data searching service search the registered metadata based on SQL and find out where the required data hosted in. Then query results then returned to the data pre-processing service.

3.2 Data Pre-processing

Before running the SYNTAM AOT retrieval program, the input data must be provided. There are 16 input parameter files in our program. The format of the file we used is ASCII. The files include calibrated bands of reflectance, sensor zenith and solar zenith. So after finding the required MODIS data, we should transfer them to the correct format to use. This process is called pre-processing. It consists of five steps, calibration, geo-reference, merging and clipping, format transfer. Calibration is used to transfer the DN value to the physical value. Geo-reference is the alignment of an image to a map so that the image has correct spatial location and orientation. The spatial information is the base of merging and clipping. Some times the queried MODIS data consist of several tracks. In that case, the data should be merged then derive the overlay region of the Terra and Aqua data.

Geo-referencing of MODIS data is time consumable and computationally intensive. Combined with the calibration, the geo-reference task is submitted to the Grid. The algorithms and the partition strategy can be found in the paper from Hu *et al.* (2005).

The merging, clipping and format transfer are combined to a unity one. When it concerns to regional or global scale, the partition strategy must be considered if there are no high-powered computer that could handle the merging process in the Grid pool. We apply dynamic filling methods to fulfil the task. Firstly, the request range is divided into regular pieces according to the available computers' amount. The sub range information and the geo-referenced data are sent to the job nodes in the grid environment. Secondly, the job nodes search the data within the specified range and fill the data into the correct location. After the required 16 parameter files are ready on the job node, the SYNTAM AOT services start up.

3.3 Job Management

The task is partitioned into many sub jobs and the jobs are identified by unique Grid job identifiers. The job manager monitors the job status from submission to completion. The job status includes running and idle. An idle job is a job, which has just been submitted into the grid pool and is waiting to be matched with an appropriate computing element or a job, which has vacated and has been returned to the grid pool. A running job is a job, which is making active progress. Finished status is reached whenever user retrieves all the output files produced by a job. The job is check-pointed for later restart. When a chosen resource refuses to accept the job, the job is vacated and waiting for the manager to reallocate to other computing element.

3.4 Post-processing

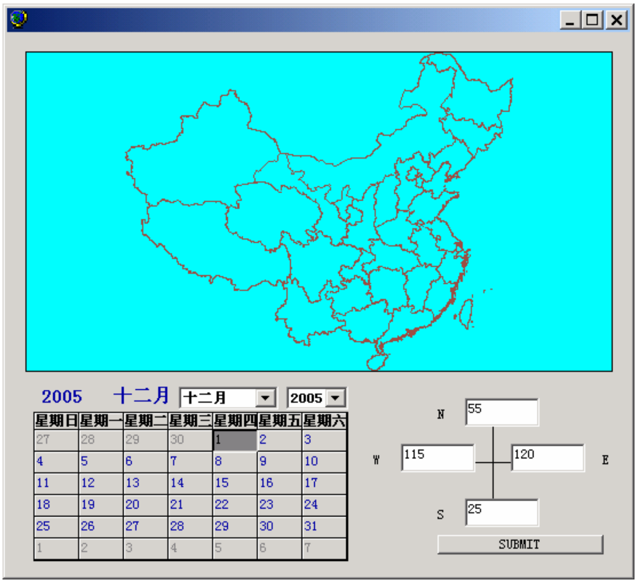
The Grid manager collects all the results returned from the job nodes. The results are merged dynamically. After all of the results are merged, the merged files are transformed to the format, which the user required and then the transformed files are finally transferred to the user.

4 Experiments

The Grid-computing environment we used is the High-Throughput Spatial Information Processing Prototype System (HIT-SIP) developed by Institute of Remote Sensing Applications, Chinese Academy of Sciences. Test data are MODIS level 1B products, which acquired from the MODIS data sharing platform (http://www.nfiieos.cn). The file format of the data is HDF format.

The files are registered to the Grid pool firstly. The metadata describes the data's quality, scope, run-across time, satellite name, unit name and provider's name. Then register the AOT retrieval services to the grid pool.

The AOT retrieval portal is shown in Figure 3. User can input the latitude and longitude or draw rectangle in the map by mouse to define the scope. The portal provides a calendar control to select the date. The experiments select different scope to test. The scope is confined to China's district. Some of test results are shown Figure 3.



2005 十二月 十二月 2005

星期日	星期一	星期二	星期三	星期四	星期五	星期六
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31
1	2	3	4	5	6	7

N 55

W 115

E 120

S 25

Fig. 3. The portal of AOT retrieval from MODIS dara

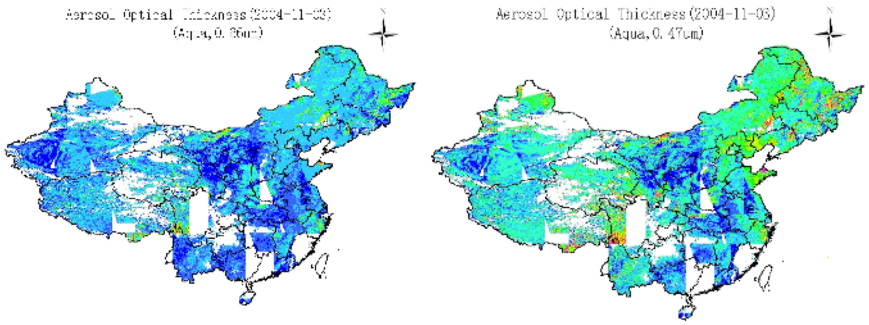

Aerosol Optical Thickness(2004-11-03)
(Aqua,0.86um)
N
Aerosol Optical Thickness(2004-11-03)
(Aqua,0.47um)
N

Fig. 4. Aerosol optical thickness results retrieved from MODIS data

5 Conclusions

In this paper, the implementation of the AOT retrieval Grid services is discussed and tested. Our tests are based on the HIT-SIP grid environment. The experiments are successful but there are some aspects we should improve in the future. One of them is the load balance. Our partition strategy doesn't consider the difference computability among the computing elements. When the job is submitted to low computability one, the whole efficient will be affected. Otherwise, when there are high power computers in the grid pool, it may be more efficient to submit most of jobs to them. A scheme should be added in order that able person should do more work. Another aspect we should improve is the data management. In our experiments, the database is file based. When the data are centralized in one node, the transferring way will be jam-packed In the future the distributed database should be build with the dynamic replica scheme to reduce the pressure on the data source nodes.

Acknowledgement

This publication is an output from the research projects "Aerosol fast monitoring modelling using MODIS data and middlewares development" (40471091) funded by NSFC, China, “Monitoring of Beijing Olympic Environment Using Remote Sensing” (2002BA904B07-2) and “863 Program - Remote Sensing Information Processing and Service Node” (2003AA11135110) funded by the MOST, China and "Digital Earth" (KZCX2-312) funded by Chinese Academy of Sciences, China.

References

Aloisio, G., Cafaro, M,2003. A dynamic Earth observation system. Parallel Computing. Vol. 29, No.10 ,pp1357-1362.

Aloisio, G., Cafaro, M.:,Epicoco, I., Quarta, G,2004. A problem solving environment for remote sensing data processing. In Proceeding of ITCC 2004: International Conference on Information Technology: Coding and Computing held in Las Vegas, NV, USA on 5-7 April 2004, (Vol.2) 56-61.

Cai G.Y., Xue Y., Tang J. K., Wang J. Q., Wang Y. G., Luo Y., Hu Y. C., Zhong S. B., Sun X. S., 2004, Experience of remote sensing information modelling with grid computing. *Lecture Notes in Computer Science,* Vol. 3039, pp.1003-1010.

Cannataro, M., 2000, Clusters and grids for distributed and parallel knowledge discovery. Lecture Notes in Computer Science, Vol. 1823, 708-716, 2000.

Hu Yincui, Xue Yong, Tang Jiakui, Zhong Shaobo, Cai Guoyin, 2005, Data-parallel Georeference of MODIS Level 1B Data Using Grid Computing. Lecture Notes in Computer Science, Vol. 3516, pp883-886.

Running, S. W., Justice, C. O., Salomonson, V. V., Hall, D., Barker, J.,Kaufman, Y. J., Strahler, A. H., Huete, A. R., Muller, J.-P., Vanderbilt,V., Wan, Z. M., Teillet, P., & Carneggie, D. (1994), Terrestrial remote sensing science and algorithms planned for EOS/MODIS. International Journal of Remote Sensing, 15(17), 3587– 3620.

Tang Jiakui, Xue Yong, Yu Tong, Guan Yanning, Cai Guoyin, Hu Yincui, 2005, Aerosol Optical Thickness Determination for Land Surface from MODIS data. Science in China (Ser. D Earth Sciences), Vol. 35, Issue 5, Pages 1-8.

Tang Jiakui, Xue Yong, Yu Tong, Guan Yanning, 2005. Aerosol Optical Thickness Determination by Exploiting the Synergy of TERRA and AQUA MODIS (SYNTAM). Remote Sensing of Environment, Vol. 94, Issue 3, Pages 327-334

Wang J. Q., Sun X. S., Xue Y., Hu Y. C., Luo Y., Wang Y. G., Zhong S. B., Zhang A. J., Tang J. K., Cai G. Y., 2004, Preliminary study on unsupervised classification of remotely sensed images on the Grid. *Lecture Notes in Computer Science,* Vol. 3039, pp.995-1002.

Revocation Scheme for PMI Based Upon the Tracing of Certificates Chains[*]

M. Francisca Hinarejos and Jordi Forné

Technical University of Catalonia (UPC),
Department of Telematics Engineering (ENTEL),
1-3 Jordi Girona, C3 Campus Nord (Barcelona), 08034 Spain
{mfcampos, jforne}@entel.upc.edu

Abstract. Public Key Infrastructure (PKI) and Privilege Management Infrastructure (PMI) can respectively be used to support authentication and authorization in distributed scenarios. The validation of certificate chains is a critical issue in both infrastructures, because it requires several costly processes, such as certificate path discovery, validation of each certificate, and so on. The problem becomes even worst in devices with limited resources (battery, memory, computational capacity, etc.) as mobile devices. In this paper we present an architecture that reduces the communication and computational overhead of certificate status checking in a complete certificate chain. The proposed tracing of the certificates chains is based on a cascade certificate revocation policy.

1 Introduction

Authentication is the assurance that the communicating entity is the one that claims to be, while authorization is the process to verify that and entity has enough rights to accede to the requested resources. Traditional access control relying on authentication and enumeration of subjects in Access Control Lists (ACLs) needs to keep the ACL consistent and up-to-date, which is difficult and present important scalability problems in distributed environments, especially when facing multidomain trust and policies

To overcome these problems, the ITU-T defined in [1] a new type of certificate, the attribute certificate (AC). Whereas a Public Key Certificate (PKC) binds an identity with a public key, an AC binds an identity with a set of attributes. A Privilege Management Infrastructure (PMI) is a collection of Attribute Certificates (ACs), with their issuing Attribute Authorities (AAs), subjects, relying parties and repositories. Through the PKIX group, the IETF is also adapting attributes certificates for authorization in the Internet.

[*] This work has been supported by the Spanish Research Council under the project ARPA (TIC2003-08184-C02-02) and the European Union funded project UBISEC. We also thank both the *Departament d'Universitats, Recerca i Societat de la Informació* and European Social Funds that support M. Francisca Hinarejos's PhD work.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1098 – 1106, 2006.

A PMI allows the following:

- *Support for distributed (role-based) access control*: With ACs, it is possible to bind an identifier and a set of attributes that can describe rights or privileges, making ACLs not necessary.
- *Support for delegation of rights*: Subjects can delegate their permissions to other subjects with no interaction with the authority during the delegation, allowing more decentralized authorization schemes.

PMI based authorization implies the verification of a single or a complete attribute certificate path (which is called *delegation path*). The checking process includes the revocation status checking, a costly process with important scalability problems. In fact, the complexity is much higher than certificate status checking in a PKI, because in a PMI it involves the validation of both the ACs and the PKC chains associated to any AC holder into the delegation path.

In this paper, we propose a mechanism that facilitates the status checking of a delegation path for a X.509 PMI. A revocation policy in cascade is used to reduce the complexity of the status checking when validating a delegation path. The main idea is quite simple: when a certificate C is revoked, all the certificates dependents on the certificate C are revoked as well. This knowledge allows the status checking on the last certificate in the path (the certificate to validate) allows to check the status of the whole path. In order to update the revocation information we use the tracing of the certificates chains.

The rest of the paper is structured as follows. In section 2, the different types of certificates path, the relationship between the certificates and the problems related to the certificates revocation are explained. The architecture, the operation and an analysis about the proposed system are explained in section 3. Then, the system is analysed in section 4. Finally, section 5 concludes.

2 State of the Art

Figure 1 shows the PMI general architecture, consisting of four main modules: 1) the AA is in charge of issue, renew, revoke and publish information related to attribute certificates, 2) the user who request access to the resources, 3) the directory storages the attribute certificates and the revocation information, 4) the privilege verifier manages the access to the resources based on the user AC.

X.509 [1] defines two types of paths: a) the certification path, which consist of the public key certificates, and b) the delegation path, which consist of the attribute certificates. The problem of validating the delegation chains is more complex than the one of verifying the certification chains. This fact is due to the verification of delegation chains involving at least the validation of the certification chain linked to each AC in the delegation path, see Figure 2.

Figure 2 shows the relations between the different certificates in the delegation path validation. The validation involves the following main steps:

- Get the PKC bound to the presented AC.
- Get and validate the certification path associated to the PKC.

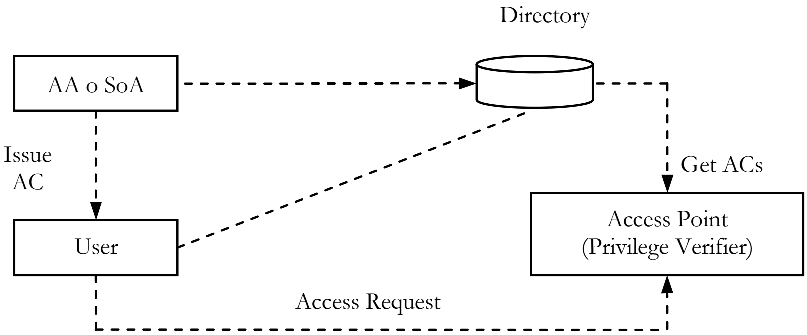


Fig. 1. General architecture of a privilege management infrastructure defined by the ITU-T in X.509 [1]

- Verify the AC signature. The privilege verifiers get the PKC of the attribute authority which conveys the public key necessary to achieve this process.
- Get the next AC in the path. The two last steps must be repeated until the AC SoA[1] is achieved. In the worst case, each certification path involved could belong to different CA domains [10]. This fact must be taken into account because it increases the complexity and the burden in the system.

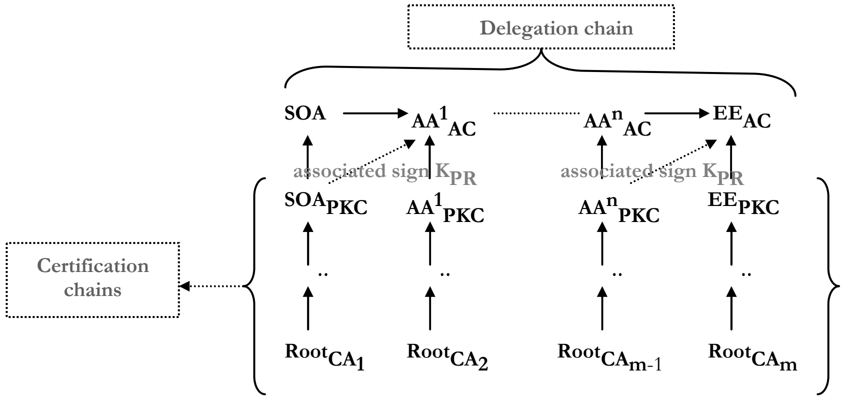


Fig. 2. Certificates chains involved in the delegation general mode defined by the ITU-T in the X.509 Recommendation [1]

2.1 Privileges Revocation

A certificate is revoked when some privileges on the certificate are no longer valid for some reasons, such as fraudulent use of the privileges by the user. If for any reason an AA revokes an AC, the other entities must realize that the revocation has occurred so they do not use an untrustworthy certificate. The revocation involves two different processes:

[1] Source of Authority (SoA) in PMI. The SoA is the equivalent to the root CA in PKI.

- The steps to take to revoke a privilege or certificate carrying the privilege.
- The mechanisms used to check the certificate status (CRL [1], OCSP [12], CRT [13], etc.). Those mechanisms try to reduce the trust on an on-line authority or the bandwidth needed to get the revocation information. However, they are not focused on the possible effects of the certificate revocation on the associated certificates.

We focus our work on the rules that establish the steps to take when a certificate in a certificates path is revoked. More specifically, our work is focused on the effect of the certificate revocation on the rest of the certificates path. We consider a cascade propagation of revocation, a possibility suggested in [3].

2.2 Related Work

The solutions presented until now are focused to reduce the burden in validation process on end entities. These solutions are based on the certificates management both local and temporal [8] [14]. However, the knowledge obtained during the checking process is used to local benefit, not for global benefit. That is, once an entity verifies a certificates path, if this path is not valid, due to a revocation of any certificate in the path, then this path will not be valid in the future. For this reason, it should be not necessary that any other entity verifies the certificates path again, or verifies another certificates path including the revoked certificate.

Another solution is to delegate the privileges to a third entity [9]. In this case, the burden to verify the certificates path moves from an entity A to another entity B. Therefore, the entity B checks the certificates path on behalf of the entity A. This process is suitable when the entity B has more resources than the entity A.

3 Proposed System

We propose to use the revocation information in a global area which is based on the cascade revocation. This knowledge will allow the status checking on a certificate to check the status of all the certificates paths involved, either the certification path or the delegation path. Also, it will allow to solve several problems, such as: avoid the no authorized privileges retention, take into account the revocation both transitive and selective, among others. These tasks are carried out through the tracing of the certifiable chains.

Next, we present the features of the proposed solution that allows carrying out the process above indicated.

3.1 Architecture

Figure 3 depicts the six possible elements that integrate a privilege management infrastructure based on X.509 attribute certificates. Next, we explain the functionalities of each module including the proposed module:

- AA/SoA (*Attribute Authority/Source of Authority*): entity in charge of issuing and revoking the attribute certificates.
- User: end entity that tries to access the resources, presenting both a PKC and an AC. This entity can revoke its certificates.

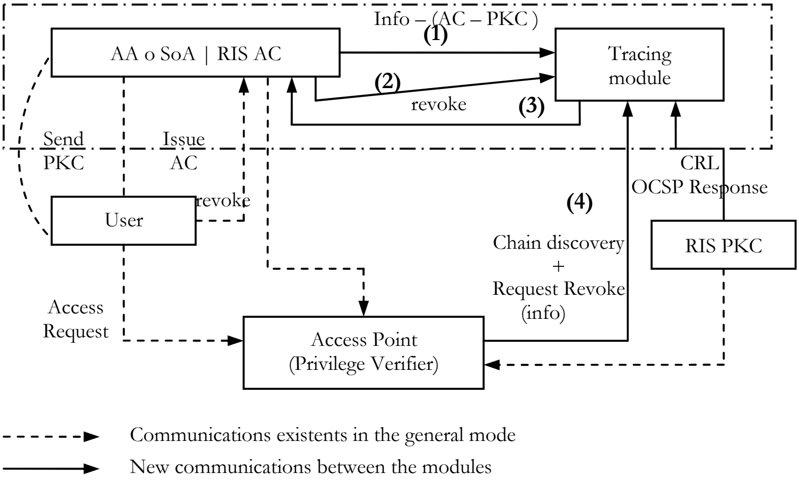

Info – (AC – PKC)
(1)
AA o SoA | RIS AC
Tracing module
(2)
revoke
(3)
Send PKC
Issue AC
User
revoke
(4)
CRL
OCSP Response
RIS PKC
Chain discovery + Request Revoke (info)
Access Request
Access Point (Privilege Verifier)
Communications existents in the general mode
New communications between the modules

Fig. 3. Tracing Module integration into the complete architecture for a revocation system in a privileges management system based on X.509 attribute certificates

- RIS-AC (*Revocation Information Server* – AC): entity in charge of receiving the revocation requests, verifying the information and creating the structure containing the certificates status information such as CRL, OCSP, etc.
- RIS-PKC *(*Revocation Information Server – PKC): this system has the same functionalities than the RIS-AC module. However it manages the public key certificates instead of attribute certificates.
- Tracing module: entity in charge of tracing the attribute certificates status. Its main task is to obtain information about the status of the different certificates: 1) the attribute certificates involved in the delegation chain, and 2) the certificates included in the certification[2] paths which are necessary to verify each AC signature in the delegation path. To carry out this task, it achieves a tracing on the certificates chains. Also, the module updates the transitive and selective effect on the delegation path.
- Privilege Verifier or Access Point to Resources: entity in charge of controlling the access to the resources. To carry out this task, the module verifies the user privileges brought on attribute certificates. The entity gets the necessary information from the different directories to take a decision. Also, it can achieve support tasks to the tracing module.

3.2 Module Operation

Figure 3 and Figure 4 depicts the procedures made by the tracing module. Next, we explain in more detail each procedure:

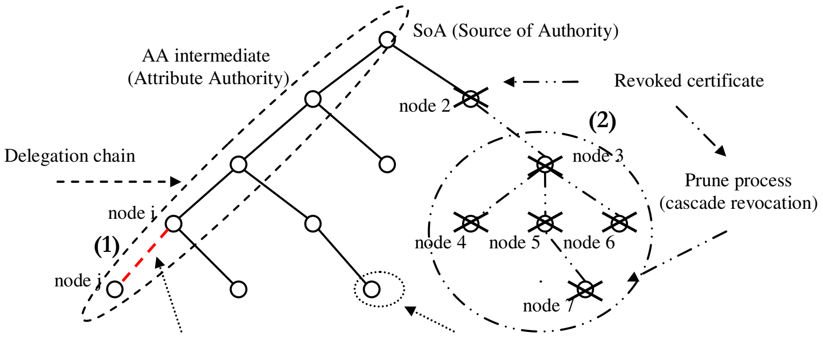
[2] This paper is focused on the delegation chains which are only built with attribute certificates.

(1) The module receives information about the delegation achieved by the authorized entities. This information is used to build the tree (see Figure 4). The information related to each attribute certificate issued is stored in a node. The dependent nodes (nodes, 4, 5 and 6) depict the ACs issued by the AC holder depicted in the upper node (node 3). The tracing is achieved on the nodes that represent the delegation chain, from the final node to the SOA node $< cert_{AC^i}, cert_{AA_1}, ..., cert_{AA_m}, cert_{SoA^i} >$. When an AA delegates any privilege to a user, the AA validates the user identity through the user PKC. This information can be stored to be used in the process of tracing the certification chains associated to the ACs.

(2) There are two possibilities to update the tree:
   a. Pruning its branches following a revocation policy in cascade. That is, when a certificate is revoked, the node that represents the certificate to revoke is searched (node 3). Both the certificate to revoke (node 3) and the dependent nodes (4, 5, 6, and 7) are eliminated in the tree. In this case, the dependent nodes are depicted by the leaves with source branch at the node to eliminate (node 3).
   b. When the validity period of the certificate has expired. In this case, the subtree with the root in the expired certificate is pruned.

(3) Make a revocation request for each certificate located under the revoked certificate. Send the request to the appropriate entity in function of the revocation mechanism used. This fact allows the coexistence with entities that not support the proposed mechanism. The process is direct and does not need to make connections to external servers if the node is internal to an AA. If the module is used as a revocation mechanism, it is not necessary to make extra processes.

(4) When a user needs to access the resources, it sends a request to the Privilege Verifier. The Privilege Verifier verifies the attribute certificate presented by the user. The Privilege Verifier requests information from the tracing module to carry out this process:



Each branch depicts the issued certificate from node i to node j

Each leaf depicts the entity that receives the attributes

Fig. 4. Tree depicting the relationships between the issued attribute certificates

a. The module searches the certificate in the tree and tries to find a delegation path for the certificate. If the module does not find information about the certificate object, the module returns an unknown code indicating an unusual situation. Therefore, the entity should deny the access to the user.
b. The certificate is valid if the module finds an entry in the tree. As a node exists, it implies that the certificate is not revoked. Therefore, the rest of the certificates involved in the path are neither revoked and the certificates have not expired yet.

4 Analysis

Next, we present a short analysis about several issues about the proposed systems, such as: the issues that influence in the implementation cost and scalability.

Issues in the implementation cost. Mainly, we study the cost on the system from two points of view:

- Volume of information to store by the module. It is only necessary to store a part of the certificate information, due to the complete certificate can be stored in any LDAP directory. Basically, the necessary information for each node in the tree is: certificate serial number, the certificate holder and the *IssuerSerial* field in the PKC bound to the AC.
- Tree update. The tree is updated: 1) when an AA issues a new AC, 2) an authorized entity revokes an AC, or 3) when the certificate has finished its validity period. The tree update will depend on the tree size and the tree search algorithm. There are different tree search algorithms such as: BFS (Breath-First Search), DFS (Depth-First Search), DLS (Depth-limited Search), IDS (Iterative Deepening Search), etc. The time and space complexity[3] of BFS is exponential with the depth of a shallowest goal node. However the DFS space complexity is linear with the maximum length of any path in the tree, but it is exponential in time complexity. On the other hand, IDS is the preferred search when search space is large and the depth of the solution is not known.
  However, the system could use additional information to search the goal node. For example, attribute certificate could convey information about the delegation path. In this case, when the module search the related information knows a priori the level where to start the search on the tree. In this case, both the time and space complexity is reduced to the maximum number of successors of any node.

Scalability. When the number of nodes number into the tree increases, scalability issues has to be considered. The problem may be alleviated by dividing the tree among different domains. Each domain is in charge of the attribute certificates under her authority. Figure 5 depicts the tracing certificates division into different domains:

(1) The main tree is managed by the SoA or by an authorized entity.
(2) The node and leaves depict the Authorities who manage its own domain. In Figure 5, the node (a) manages the delegations make into the Domain 1.

[3] Time complexity: how long does it take to find a solution?. Space complexity: how much memory is needed to perform the search?

(3) The certificate revocation into a domain, only affect into the domain. That is, the revocation has not effect over the other domains.

(4) The certificate revocation of an AA who manages a domain involves the revocation of the complete tree into the domain.

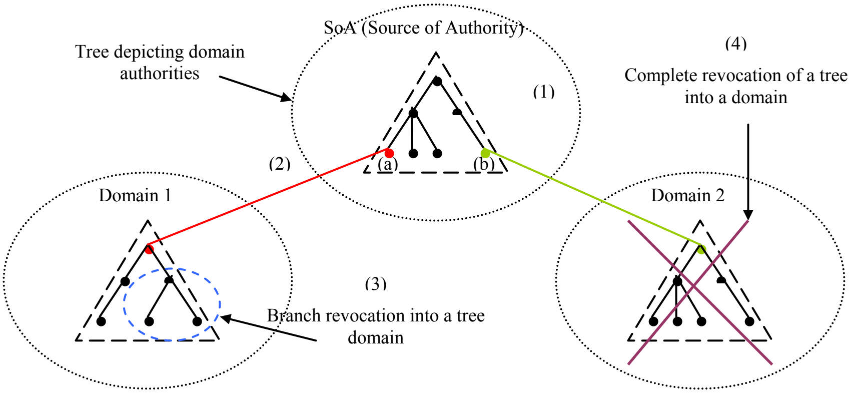

SoA (Source of Authority)
Tree depicting domain authorities
(1)
(4)
Complete revocation of a tree into a domain
(2)
(a)
(b)
Domain 1
Domain 2
(3)
Branch revocation into a tree domain

Fig. 5. Tree management divided into different domains. Each domain manages a subset of the attribute certificates under its domain.

5 Conclusions

In this paper we have presented the procedures and necessary elements to implement a privileges revocation policy in cascade, based on the tracing of the certificates chains.

There are scenarios where the burden on the tracing module can be considerable due to the great number of nodes to manage. We have proposed to manage different trees by different domains authorities. Each domain authority updates his tree and it communicates the revocations to the SoA domain. So the global revocation information is approachable by all domains.

Although the solution has been presented as a solution to delegation paths status checking, it can also be used in other scenarios. For example, it can be used to do a tracing of the certification chains in each PKI domain, allowing the global system to become more scalable. The module location in the PMI architecture is an important issue, and its location will depend on the benefits or constraints in the environment.

Our solution is especially suitable to scenarios where the end entity has limited resources - memory, computational capacity and bandwidth -, such as mobile devices.

References

1. ITU-T Recommendation X.509, *Information technology – Open Systems Interconnection – The Directory: Public Key and Attribute Certificate Frameworks*. 2000
2. IETF RFC 3281, S. Farrell.R. Housley, *An Internet Attribute Certificate Profile for* Authorization. April 2002.

3. Hagstrom, A.; Jajodia, S.; Parisi-Presicce, F.; Wijesekera, D., *Revocations –a classification*. Computer Security Foundations Workshop, 2001. Proceedings. 14th IEEE, 11-13 June 2001. Page(s): 44 -58.
4. B. Sadighi Firozabadi and M. Sergot, *Revocation in the Privilege Calculus.* In Proceedings of the 1st International Workshop on Formal Aspects in Security and Trust (FAST 2003), pages 39-51, September 2003.
5. Popescu, B.C.; Crispo, B.; Tanenbaum, A.S, *A certificate revocation scheme for a large-scale highly replicated distributed system*. Computers and Communication, 2003. (ISCC 2003). Proceedings. Eighth IEEE International Symposium on , 2003 Page(s): 225 -231.
6. H.Khurana and V.D. Gligor, *Review and Revocation of Access Privileges Distributed with PKI Certificates*. 8th International Workshop on Security Protocols, LNCS, Vol. 2133, Springer-Verlag, 2000, pp 100-124.
7. RFC 2510. C. Adams, Entrust Technologies, S. Farell, SSE, *Internet X.509 Public Key Infrastructure Certificate Management Protocols*. March 1999.
8. Yki Kortesniemi, *SPKI Performance and Certificate Chain Reduction,*Informatik 2002, Workshop "Credential-basierte Zugriffskontrolle in offenen, interoperablen IT-Systemen", Dortmund, 30.9. - 3.10.2002.
9. RFC 3379. D. Pinkas, Bull, R. Housley, RSA Laboratorios, *Delegated Path Validation and Delegated Path Discovery Protocol Requirements*. September 2002.
10. Steve Lloyd, PKI Forum, *Understanding Certification Path Construction*. White Paper. September 2002.
11. Yassir Elley, Anne Anderson, Steve Hanna, Sean Mullen, Radia Perlman, Seth Proctor, *Building Certification Paths: Forward vs. Revers*e. Network and Distributed System Security Symposium Catamaran Resort Hotel San Diego, California. 8-9 February 2001.
12. RFC 2560. Myers M., Ankney R., Malpani A., Galperin S., and C. Adams, *X.509 Internet Public Key Infrastructure – Online Certificate Status Protocol – OCSP*. June 1999.
13. P. C. Kocher, *On Certificate Revocation and Validation.* Proceedings of the 2nd International Conference Financial Cryptography, 1465 of LNCS, pp. 172-177, Springer, 1998.
14. Selwyn Russell, Ed Dawson, Eiji Okamoto and Javier Lopez, *Virtual certificates and synthetic certificates: new paradigms for improving public key validation*. Computer Communications, Volume 26, Issue 16, 15, Pages 1826-1838.

Nailfold Capillary Microscopy High-Resolution Image Analysis Framework for Connective Tissue Disease Diagnosis Using Grid Computing Technology*

Kuan-Ching Li[1], Chiou-Nan Chen[1,2], Chia-Hsien Wen[1],
Ching-Wen Yang[3], and Joung-Liang Lan[4]

[1] PDPC - Parallel and Distributed Processing Center,
Department of Computer Science and Information Management,
Providence University Shalu, Taichung 43301, Taiwan
{kuancli, cnchen, chwen}@pu.edu.tw
[2] Laboratory of Bioinformatics and Computational Biology,
Department of Computer Science,
National Tsing Hua University, Hsinchu 30013, Taiwan
[3] Computer and Communication Center,
Taichung Veterans General Hospital, Taichung 40705, Taiwan
cwyang@vghtc.gov.tw
[4] Department of Internal Medicine,
Taichung Veterans General Hospital, Taichung 40705, Taiwan
jllan@vghtc.gov.tw

Abstract. Nailfold capillary microscopy examination has been used since late 1950s as a non-invasive in-vivo technique for diagnosing and monitoring connective tissue disease in adults. Disorders such as Primary Raynaud's phenomenon, progressive systemic sclerosis, and rheumatoid arthritis were detected in more than 80% of adult patients, by analyzing such high resolution images. Internet computing and grid technologies promise to change the way we tackle complex problems. Grid computing environments are characterized by interconnecting a number of heterogeneous hosts in geographically distributed domains. They enable large-scale aggregation and sharing of computational, data and other resources across institutional boundaries. In this paper, we discuss and develop a framework for nailfold capillary microscope image acquisition and analysis, using computational power provided by grid platforms. In this way, not only useful medical information can be extracted from large amount of history anamneses in an efficient way, with the use of a number of adequate techniques and methods in high performance computing, but also to diagnose abnormal nailfold capillary in far shorter time, to diagnose patient's disease in real-time basis. Based on the results of the classification, analysis of history anamneses are done to discover updated health information possibly hidden in patients' medical records.

Keywords: Nailfold Capillary Microscopy, Medical Image, Image Processing, Pattern Recognition, Grid Computing.

* This research is supported in part by National Science Council, Taiwan, under grant no. NSC94-2213-E-126-005, and Taichung Veterans General Hospital and Providence University Collaborative Project, under grant no. TCVGH-PU-958107.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1107 – 1115, 2006.

1 Introduction

Analysis of nailfold capillaries microscopy images have been used since late 1950s, and accepted as a non-invasive in-vivo technique to diagnose adults with connective tissue diseases. By using this technique, disorders such as Primary Raynaud's phenomenon, progressive systemic sclerosis, dermatomyositis, and rheumatoid arthritis and other related abnormalities can be detected in more than 80% of adult patients, while rheumatic diseases in most children patients. Such technique has proven its efficiency to provide necessary information to aid the diagnosis of a number of disorders both in adults and children.

A number of previous techniques for diagnosing process are widely available. Some of them have relied on measuring the capillary loop dimensions from single video frames. Unfortunately, the major drawback is that the loops can appear incomplete at any one instant, since the capillary walls themselves are transparent and there may have gaps in the flow of red blood cells [4]. An alternative approach due to this drawback is a method in which several video frames from a sequence can be integrated into a single image, averaging the temporal variability and thus, to build up a "mosaic" of the whole area under investigation in much high resolution than previous approach. The detailed registration process, its robustness and accuracy of this method have already been discussed in [1, 2, 3]. Though, there is no computer based system available to identify whether such high resolution image is a normal or abnormal nailfold capillary.

Grid computing is the most important computer and network technology recently available. A computational grid is a collection of distributed and heterogeneous computing nodes that has emerged as an important platform for computation intensive applications. They enable large-scale aggregation and sharing of computational, data and other resources across institutional boundaries. It offers an economic and flexible model for solving massive computational problems using large numbers of computers, arranged as clusters embedded in a distributed infrastructure.

In this paper, data acquisition and analysis computing system framework are described, based on the use of grid technology, in order to obtain list of possible diseases associated with high resolution images provided. These high resolution images are obtained via optical microscope, and a widely used technique today is to measure the size of the capillaries at the base of the fingernail (nailfold) to diagnosis patient's health and possible associated diseases.

The remaining of this paper is organized as follows. In section 2, background of nailfold capillary microscopy images and their classification are introduced, while in section 3 the motivation and some previous related researches are presented. In section 4, the framework of the proposed computer system is discussed, from microscopy image acquisition to image recognition. In addition, it is introduced the grid computing platform to be used for image analysis and recognition computations, since such high resolution images demands large amount of computational cycles. Finally, in section 5, some conclusion remarks about the proposed framework and some future works in this topic are discussed.

2 Background

Capillaries are the thinnest blood vessels and form a microcirculation that links the arterioles and venules; the diameter is so small that red blood cells can only pass through singly. Capillary walls are formed from a layer of tissue so thin that waste products, nutrients and gases can be exchanged between the body's tissues and blood. The vessels of the subpapillary plexus run approximately parallel to the surface of the epidermis, and the capillary loops grow from these vessels into the dermal papillae and normal to the surface. Each of the papillae contains one capillary that may be seen under strong illumination with a microscope. In the nailfold area, they may appear as a tiny red point that corresponds to the top of the capillary loop and, as the capillary loops become progressively parallel to the skin surface, they appear more and more loop-like. A normal capillary comprises an arterial (afferent) and a venous (efferent) limb connected by the apical part.

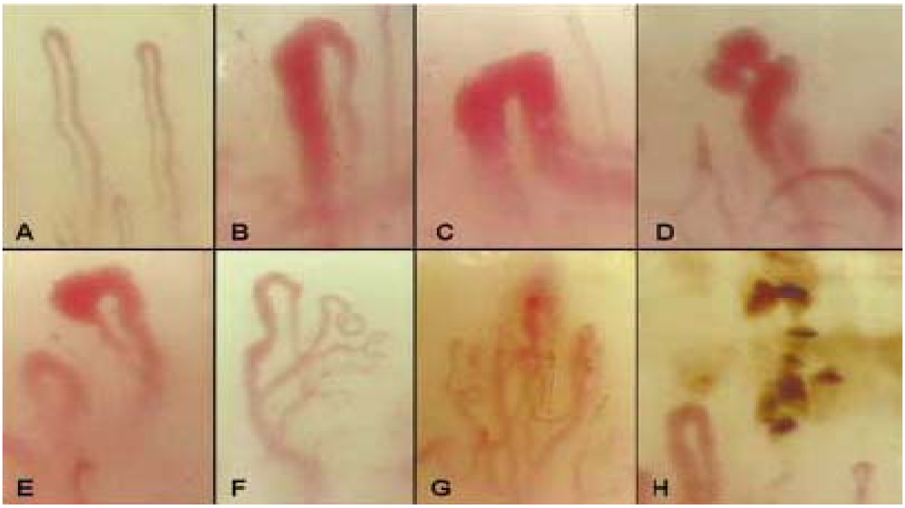


Fig. 1. Nailfold capillaroscopy images: (A) Normal capillaries, (B) Homogeneously enlarged loops of the efferent limb, (C) Megacapillary, (D) Tortuous and enlarged loop, (E) Irregularly enlarged loop, (F) Budding of capillaries, (G) Bushy capillaries, and (H) Capillary hemorrhages

Figure 1 shows the normal and abnormal nailfold capillaroscopy images. The normal capillary landscape is a uniform palisade of loops. There are several classes of abnormal nailfold capillaries, and they are listed as: Homogeneously enlarged loops of the efferent limb, Megacapillary, Tortuous and enlarged loop, Irregularly enlarged loop, Budding of capillaries, Bushy capillaries, and Capillary hemorrhages. The shape of each patient's capillaries may remain unchanged for many years, though it shows a tendency to become tortuous and dilated with age. Permanent structural changes occur over long periods, when the microcirculation remodels the blood vessels by changing their length, diameter, wall thickness, tortuosity and number; such long-term changes are widely recognized as characteristic responses to certain diseases.

The skin capillaries come close to the surface of the skin and may be viewed conveniently through a microscope at the nailfold of the fingers and toes. The normal capillary microscopy is a uniform baluster with loops that are homogeneous in size and pattern. Several researchers have observed this pattern was completely disorganized in the nailfold presence in certain disease. For instance, ischaemia, diabetes mellitus, chronic venous incompetence, lympedema, Primary Raynaud's phenomena, progressive systemic sclerosis, mixed connective tissue disease, dermatomyositis, rheumatoid arthritis and vibration disease. [10, 12, 16]. Different type of abnormal nailfold capillary could relate to different diseases. Therefore, determining the abnormality of nailfold capillary is very important to clinical diagnosis.

2.1 Related Researches

Nailfold capillary microscopy has been used to diagnose a number of diseases more than 50 years. In addition to patterned abnormalities, diagnosed measurements have been made of capillary density, capillary blood flow velocity (laser Doppler fluxmetry) and the diffusion of dyes through the capillary wall [5, 15]. Some major investigations in this topic were focused on the capillary blood flow video [1, 2, 3, 6, 16] or based on the morphological taxonomy [11]. There is not available any method or technique that is able to provide nailfold capillary image diagnosis, specifically in investigations using high performance computerized technologies in medical image processing field.

Previous techniques have relied on measuring the capillary loop dimensions from single frame or a serial video frames. These systems only capture and store capillary video frames, and without further detection or diagnosis for abnormal capillary. Grid computing has been applied to many scientific problems, such as climate modeling, computational biology, military applications, among others, but only a few applications in medical or clinical informatics field. We believe that the proposed framework in this paper will be able to contribute to a novel technique collaborating medical and high performance computing field researches.

3 Motivation

The Division of Allergy, Immunology and Rheumatology, Department of Internal Medicine, Taichung Veterans General Hospital (Taichung VGH), located in Taichung City, Taiwan, is the largest authority in Taiwan when speaking of treating immunological diseases in both adults and children. Their patients are numerous across Taiwan, and they have traced and kept this large amount of anamneses for more than two decades, being one the best resources for our research. It would be impractical that physicians analyze each of high resolution images and classify them, elevating even more medical costs as overall. Therefore, there is a need for powerful image processing and recognition computer based tool for this complex and meaningful work, by returning us in fewest time the classification of given high resolution image, and possible diseases associated with these high resolution images.

4 Proposed Computer Based System Framework

High resolution nailfold capillary microscopy images are obtained from patient's exams at Taichung VGH. These images are transferred to Providence University's PDPC via data grid technology, and then, submitted for analysis in our grid platform. Once computations related to one particular patient are concluded, the computerized diagnosis is transmitted back to Taichung VGH, including it in patient's medical records. Figure 2 shows the scheme for the complete process.

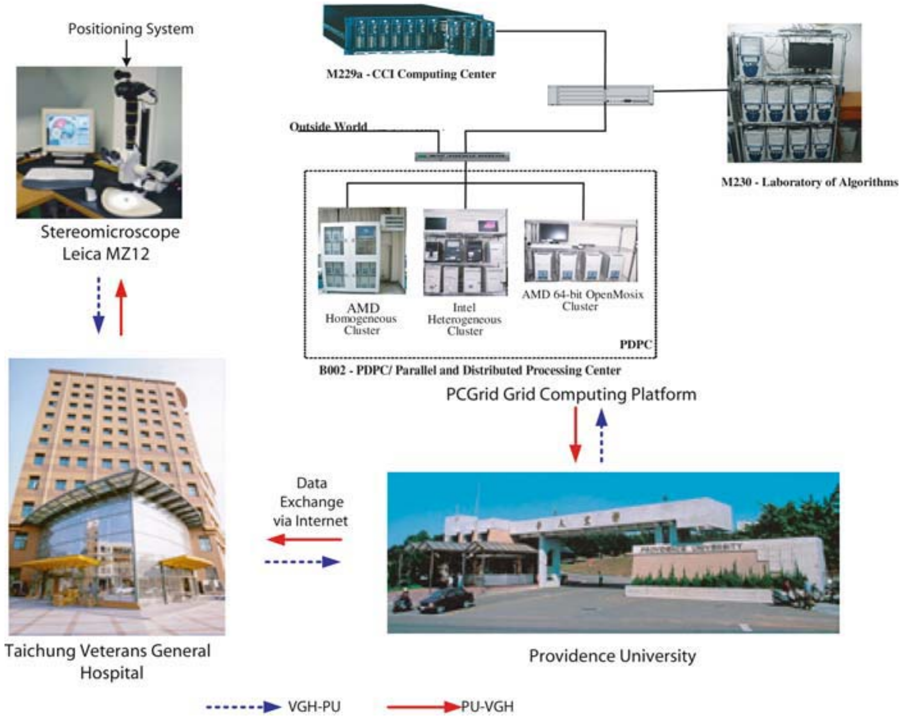


Fig. 2. Complete geographical scheme for nailfold microscopy image analysis and diagnosis

4.1 Data Acquisition System

A color digital video camera is attached to a stereomicroscope with fiber optic illumination, in order to capture nailfold capillary high resolution images from the patient. The finger to be examined is gently held in position on the microscope base plate, by the parent for child patient.

The whole nailfold is initially examined under low magnification, to determine the distribution of any obvious abnormalities. Since abnormalities are evenly distributed [9], the middle portion of the nailfold is magnified and then, photographed using the color digital camera. The set of all high resolution images belonging to the same patient are saved in a specific folder. Figure 2 shows the sequence of steps to be followed, in order to obtain a patient's high resolution nailfold capillary images.

Taichung Veteran's General Hospital (Taichung VGH) and Providence University are interconnected via TANET (Taiwan Academic Network), of 1Gbps. Therefore, high resolution images taken at Taichung VGH are transferred to Providence University's PDPC for processing via Data Grid technology.

4.2 Computational Grid Platform

We have built a grid platform named PCGrid, which stands for Providence University Campus Grid, is a computing environment built by interconnecting a number of cluster platforms currently installed in different laboratories and computing centers distributed across different floors inside College of Computing and Informatics. Such computing facility's goal is to support a number of different topics of research investigations among our college's faculties. Topics of research among our faculties that demand such high computational power include computational biology, CFD, image rendering, data distribution, parallel application design, performance evaluation and analysis, visualization toolkit, parallel application graph representation, automatic and manual computing node selection, thread migration and scheduling in cluster and grid environments, among others. Previous experiences with a number of challenging topics makes us prepared to investigate the research proposed in this paper. The Figure 3 illustrates the PCGrid grid computing infrastructure.

The PCGrid computing platform [13, 14] has a total of 39 computing nodes and 6 IBM Blade nodes, of different CPU speed and memory sizes, total storage of more than 5TB, interconnected via Gigabit Ethernet (1Gb/s). We believe that this computing platform will perfectly be suitable as research platform for the proposed investigation.

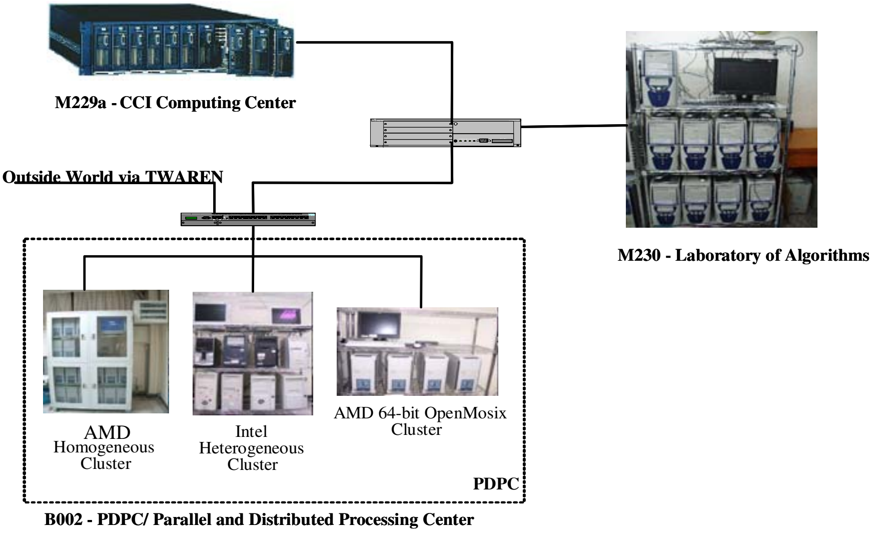

M229a - CCI Computing Center
Outside World via TWAREN
M230 - Laboratory of Algorithms
AMD Homogeneous Cluster
Intel Heterogeneous Cluster
AMD 64-bit OpenMosix Cluster
PDPC
B002 - PDPC/ Parallel and Distributed Processing Center

Fig. 3. The PCGrid grid computing platform

4.3 Method

The proposed technique and computer based system is divided into two subsystems. The former one is the image pre-processing subsystem, and the latter is content feature detection subsystem, both built over PCGrid grid computing platform.

Before recognizing the abnormal nailfold capillary, we must know where the capillary is. The image pre-processing system is used to enhance the quality of high resolution capillary microscopy images, and extract the nailfold capillary from images. We will use a number of high performance image processing and pattern recognition technologies in this subsystem; for instance: contrast enhancement, noise ignoring, edges enhancement, among others. Some previous researches have relied in statistical shape modeling [7, 8], which will also be considered for investigation and analysis of its performance.

The latter subsystem, named content feature detection subsystem is used to recognize and identify the abnormal nailfold capillaries. After high resolution image is pre-processed and edges nailfold capillary enhancement, it is possible to indicate the shape of capillary and determine whether it is abnormal. Because of large number of computational cycles available, such recognition that highly demands computational power and data intensive will definitely speedup all computational process.

One of the goals in this proposed research involves the development of a novel algorithm to detect and determine the capillary abnormality. The development of both computing system and algorithm are completely based in grid computing technology, and thus, the algorithm development and system implementation must be feasible and fulfill grid technology requirements, that is, points of concern include heterogeneous computing and network environments, analyzing computing node communication, data distribution and partitioning, performance issues, among others.

5 Conclusion

The proposed method integrates the use of grid computing technology and medical and clinical researches of nailfold capillary microscope diagnosis. The enormous amount of anamnesis is covered of useful medical information. Though, they need powerful computational resources and storage to handle it. The grid computing technology provides economic and flexible computational power, as also feasibility to process medical informatics problem such as nailfold capillary microscope diagnosis.

The proposed system included two major subsystems. The former is an image enhancement module, which is a pre-processing procedure used to obtain high resolution nailfold capillary microscopy images. The latter is a content detective module used to recognize between normal and abnormal nailfold capillaries, as also to classify the category of the abnormal nailfold capillaries. The complete system is being developed based on grid technology. We can investigate the enormous historical anamnesis of nailfold capillary microscope through this powerful computing system, in order to discover new or hidden medical information. New discoveries from these medical records will definitely help in early disease diagnosis and treatments.

As future work, search and development for suitable and computationally efficient image recognition algorithms for processing in distributed environments is listed in this first step. The data grid environment will also be investigated in parallel for its implementation, in order to transfer data among these two sites. Once successful such implementation, other medical divisions and departments of the same hospital, as also other hospitals, medical centers and laboratories can also be interconnected to such environment, to provide a faster and efficient way for public medical care.

References

[1] P.D. Allen, C.J. Taylor, A.L. Herrick, T. Moore, "Enhancement of Temporally Variable Features in Nailfold Capillary Patterns", British Machine Vision Conference 1998.

[2] P.D. Allen, C.J. Taylor, A.L. Herrick, T. Moore, "Image Analysis of Nailfold Capillary Patterns", Medical Image Understanding and Analysis 1998.

[3] P.D. Allen, C.J. Taylor, A.L. Herrick, T. Moore, "Image Analysis of Nailfold Capillary Patterns From Video Sequences", *Medical Image Computing and Computer-Assisted Intervention* 1999.

[4] P.D. Allen, V.F. Hillier, T. Moore, M.E. Anderson, C.J. Taylor, and A.L. Herrick, "Computer Based System for Acquisition and Analysis of Nailfold Capillary Images", Medical Image Understanding and Analysis, 2003.

[5] A. Bollinger, and B. Flagrell, "Clinical capillaroscopy - a guide to its use in clinical research and practice", Stuttgart:Hogrefe and Huber, 1990.

[6] M. Bukhari, S. Hollis, T. Moore, M. I. V. Jayson and A. L. Herrick, "Quantitation of microcirculatory abnormalities in patients with primary Raynaud's phenomenon and systemic sclerosis by video capillaroscopy", Rheumatology 39: 506-512, Oxford Journals, 2000.

[7] R.M. Davies, C.J. Twining, T.F. Cootes, J.C. Waterton, C.J. Taylor, "A minimum description length approach to statistical shape modelling", IEEE Transactions on Medical Imaging, vol. 21, p. 525-537, 2002.

[8] R.M. Davies, C.J. Twining, P.D. Allen, T.F. Cootes, C.J. Taylor, "Building optimal 2D statistical shape models", Image and Vision Computing, vol. 21, p. 1171-1182 , 2003.

[9] P. Dolezalova, S.P. Young, P.A. Bacon, and T.R. Southwood, "Nailfold capillary microscopy in health children and in childhood rheumatic diseases: a prospective single blind observational study", Ann Rheum Dis., 62(5), p.444-449, 2003.

[10] P. M. Houtman, C. G. M. Kallenberg, V. Fidler, A. A. Wouda, "Diagnostic significance of nailfold capillary patients with Raynaud's phenomenon", *Journal of Rheumatology*, Vol. 13, No. 3, p 556, 1986.

[11] B.F. Jones, M. Oral, C.W. Morris, E.F. Ring, "A proposed taxonomy for nailfold capillaries based on their morphology", *IEEE Transactions on Medical Imaging*, Vol. 20, No. 4, April 2001.

[12] F. Lefford, J. C. W. Edwards, "Nailfold capillary microscopy in connective tissue disease: a quantitative morphological analysis", *Anal of the Rheumatic Diseases*, Vol. 45, pp741-749, 1986.

[13] K.C. Li, H.Y. Cheng, C.T. Yang, C.H. Hsu, H.H. Wang, C.W. Hsu, S.S. Hung, C.F. Chang, C.C. Liu, and Y.H. Pan, "Visuel: a Novel Performance Monitoring and Analysis Toolkit for Cluster and Grid Environments", in *ICA3PP'2005 The 6th International Conference on Algorithms and Architectures for Parallel Processing*, LNCS 3719 Springer, M. Hobbs, A.M. Goscinski, W. Zhou (Eds.), Melbourne, Australia, p. 315-325, 2005.

[14] K.C. Li, C.N. Chen, C.W. Hsu, S.S. Hung, C.F. Chang, C.C. Liu, C.Y. Lai, “PCGrid: Integration of College’s Research Computing Infrastructures Using Grid Technology”, in *NCS'2005 National Computer Symposium*, Tainan, Taiwan, 2005.

[15] Nailfold capillary microscopy, *Archives of Disease in Childhood*, 2003;88;1004-doi:10.1136/adc.88.11.1004.

[16] A. Studer, T. Hunziker, O. Lutolf, J. Schmidli, D. Chen, F. Mahler, “Quantitative nailfold capillary microscopy in cutaneous and systemic lupus erythematosus and localized and systemic scleroderma”, *J. of American Academy of Dermatology*, vol. 24, no. 6, p. 941-945, 1991.

EFH: An Edge-Based Fast Handover for Mobile IPv6 in IEEE 802.11b WLAN[*]

Sangdong Jang and Wu Woan Kim

Division of Computer Science and Engineering, Kyungnam University,
Masan, South Korea
angong@kyungnam.ac.kr, wukim@kyungnam.ac.kr

Abstract. Mobile IPv6 has been designed to manage movements of mobile nodes among wireless IPv6 networks. Nevertheless, a mobile node cannot receive IP packets on its new point of attachment until the handover completes. Therefore, a number of extensions of Mobile IPv6 such as FMIPv6 and HMIPv6 have been proposed to reduce the handover latency and the number of lost packets. In this paper, a new mechanism based on information of *edge APs* is proposed for fast handover. The proposed mechanism provides the faster acquisition of neighboring subnet information than that of FMIPv6. In addition the information of the subnets is used more flexibly to determine L2 handover or L3 handover. Moreover, the proposed mechanism can reduce amount of traffic and the handover latency in comparison with the FMIPv6 during the handover. This research focuses on Fast Handover for MIPv6 which is an extension of Mobile IPv6 that allows the use of L2 triggers to anticipate the handover. The results of the handover latency are calculated with the L2 properties of IEEE 802.11b. In particular, we take into account the L2 handover and the L3 handover for two fast handover scenarios of the wireless networks.

1 Introduction

The fast Internet evolution with the enormous growth in the number of users of wireless technologies has resulted in a strong convergence trend toward the usage of IP as the common network protocol for both of fixed and mobile networks. Future *All-IP* networks will allow users to maintain service continuity while they move through different wireless systems.

Mobile IPv6 (MIPv6) [1] has been designed to manage movements of mobile nodes among wireless IPv6 networks. The protocol provides an unbroken connectivity to IPv6 mobile nodes when it moves from one access point of attachment to another. MIPv6 sets up a messages exchange to notify its new localization by a binding between the mobile node addresses to the correspondent(s) of a mobile node.

Nevertheless, the mobile node cannot receive IP packets on its new point of attachment until the handover finishes. This latency includes the new prefix discovery

[*] This work was supported by Kyungnam University Research Fund.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1116 – 1125, 2006.

on the new subnet, the new Care of Address configuration, and the binding update time needed to notify the new localization of the mobile node to the correspondents and the home agent. This time is called the handover latency.

Actually, the handover latency might be too long to perform real time multimedia applications. In most cases, the impact of the handover latency terribly degrades the IP stream of the mobile node. Therefore, there are many extensions of MIPv6 and new protocols [2][5] proposed to improve the IP connectivity of mobile nodes. The purpose of these proposals is for reducing the latency and the number of packets lost due to the handover between one point of attachment to another, and for reducing the signaling load on the MIPv6 home agent and on the correspondent nodes.

FMIPv6 (Fast Handover for MIPv6) [2] allows the mobile nodes to create new valid Care of Address before the movement to new wireless access point. If the protocol successfully completes, the layer3 (L3) handover latency only becomes the layer 2 (L2) handover latency.

The purpose of this paper is to reduce the latency needed by handover to move the flow of a mobile node from one access network to another. Two cases are going to be considered with regarding that mobile node receives the FBack on the current AP coverage or not. Simulations of our approach are based on IEEE 802.11b WLAN [4]. The handover latency involved in FMIPv6 and our new proposed approach is evaluated in our simulation.

In section 2 overviews of MIPv6, its extension FMIPv6 and IEEE 802.11b are presented. Then, in the following section, the proposed edge-based fast handover is presented and compared with FMIPv6. In the next section, simulation results are used to evaluate the handover latency in FMIPv6 and our new proposed approach. Finally, conclusions are given in section 5.

2 MIPv6, FMIPv6 and the Proposed Approach

In this section, we remind the handover procedure as it is defined in MIPv6 and in FMIPv6 and IEEE 802.11b WLAN.

2.1 MIPv6

MIPv6 is designed to manage mobile nodes movements between wireless IPv6 networks and is inherently optimized by using a direct notification mechanism to the nodes that know and route packets to the mobile node's new location. Every mobile node (MN) has a home network and is identified by a home IP address on that network. The 128-bit IPv6 address consists of a 64-bit routing prefix, which is used for routing the packets to the right network, and a 64-bit interface identifier, which identifies the specific node on the network and can essentially be arbitrary. Thus, IP addresses in MIPv6 can identify either a node or a location on the network, or even both. A router in the home network called a *home agent (HA)*, acts as the mobile node's trusted agent and forwards IP packets between the mobile's *correspondent nodes (CN)* and its current location, identified by the *care-of address (CoA)*. The MIPv6 protocol also includes a location management mechanism using *Binding*

Updates (BU). When a mobile node remains in its home network, it communicates with this home address like another IPv6 node with its CN. When a mobile node moves to a new point of attachment in another subnet, it can send BUs to its CNs as well as HA to notify them about the new location so that they can communicate directly. Moreover, it cannot use its home address any more to send packets in the new subnet. Therefore it needs to acquire a new valid CoA in the visiting subnet. Then, it informs its HA and its CNs about the binding between its home address and its *new Care-of Address (nCoA)*. On the other hand, the home address always identifies the communication, even if the mobile node is in a visited network.

MIPv6 describes the protocol operations for a mobile node to maintain connectivity to the Internet during its handover from one access router to another. These operations broadly involve movement detection, IP address configuration, and location update phase. Actually, the combined handover latency must be too long to perform real time multimedia applications. Throughput-sensitive applications can also benefit from reducing this latency. To reduce the service degradation that a mobile node could suffer due to a change in its point of attachment *Fast Handovers for Mobile IPv6 (FMIPv6)* has been proposed.

2.2 FMIPv6

FMIPv6 protocol describes a framework as well for the mobile-controlled handover as for the network-controlled handover with only a small difference in the messages order. In the case of the network-controlled handover, a specific entity of the network decides when the mobile node needs to move to a new point of attachment. This entity can be the *current AR (pAR)* offering the connectivity to the mobile node or a dedicated equipment in the subnet which manages the mobile node movements.

FMIPv6 enables a MN to quickly detect that it has moved to a new subnet by providing the *new access point (nAP)* and the associated subnet prefix information when the MN is still connected to its current subnet. A MN discover available access points using link-layer specific mechanisms (e.g., scan) and then request subnet information corresponding to one or more of those discovered access point. The MN may do this after performing router discovery. The *Router Solicitation for Proxy Advertisement (RtSolPr)* and the *Proxy Router Advertisement (PrRtAdv)* messages are used for aiding movement detection. Through the RtSolPr and PrRtAdv messages, the MN formulates a prospective nCoA and sends a *Fast Binding Update (FBU)* message. The purpose of FBU is to authorize pAR to bind *current Care-of address (pCoA)* to nCoA, so that arriving packets can be tunneled to the new location. The FBU should be sent from pAR's link whenever feasible.

During the IETF discussions regarding this proposal two different mechanisms have been described: Predictive and Reactive. Depending on whether a *Fast Binding Acknowledgment (FBack)* is received or not on the previous link, which clearly depends on whether FBU was sent in the first place, there are two modes of operation.

"Predictive" Fast Handover. The MN receives FBack on the previous link. This means that packet tunneling would already be in progress by the time the MN handovers to nAR. The MN should send *Fast Neighbor Advertisement (FNA)* immediately after attaching to nAR, so that arriving as well as buffered packets can be forwarded to the MN right away.

Before sending FBack to MN, pAR can verify if nCoA is acceptable to nAR through the exchange of *Handover Initiate (HI)* and *Handover Acknowledge (Hack)* messages. When stateful assignment is used, the proposed nCoA in FBU is carried in HI, and nAR may consider assigning the proposed nCoA. In any case, the assigned nCoA must be returned in HAck, and pAR must in turn provide the assigned nCoA in FBack. If there is an assigned nCoA returned in FBack, the MN must use the assigned address (and not the proposed address in FBU) upon attaching to nAR. The HI and HAck protocol exchange to verify nCoA acceptability.

"Reactive" Fast Handover. The MN does not receive FBack on the previous link. One obvious reason for this is that the MN has not sent the FBU. The other reason is that the MN has left the link after sending the FBU (which may be lost) but before receiving an FBack. Without receiving an FBack in the latter case, the MN cannot ascertain whether PAR has successfully processed the FBU. Hence, it (re)sends an FBU as soon as it attaches to NAR. In order to enable NAR to forward packets immediately (when FBU has been processed) and to allow nAR to verify if nCoA is acceptable, the MN should encapsulate FBU in FNA. If nAR detects that nCoA is in use when processing FNA, for instance while creating a neighbor entry, it must discard the inner FBU packet and send a Router Advertisement with *Neighbor Advertisement Acknowledge (NAACK)*option in which nAR may include an alternate IP address for the MN to use. This discarding avoids the undesirable outcome of address collision, even though the chances of such a collision are extremely low. A FMIPv6 handover nominally consists of the following messages :

a. The MN sends a RtSolPr to find out about neighboring ARs.
b. The MN receives a PrRtAdv containing one or more [AP-ID, AR-Info] tuples.
c. The MN sends a FBU to the pAR.
d. The pAR sends an HI message to the nAR.
e. The nAR sends a HAck message to the pAR.
f. The pAR sends a FBack message to the MN in the new link. The FBack is also optionally sent on the previous link if the FBU was sent from there.
g. The MN sends FNA to the nAR after attaching to it.

2.3 Fast Handover in IEEE 802.11b WLAN

IEEE 802.11b enables two operational modes. The first one is the ad hoc mode where there is no central point. The stations communicate directly if they can hear each other. The second is the infrastructure mode, where all the communications occur via an access point. An access point is a dedicated equipment which has at least one wireless interface and one wired interface. It is a bridge between the wired network and the wireless LAN. The communications occur within the cover area of the access point. One or more mobile nodes connected to an access point are called a BSS and several BSS connected together through an Ethernet link under the same subnet are called an ESS (Extended Service Set).

When several mobile nodes are connected to an access point, they must share the channel access. IEEE 802.11b defines two access methods: the basic protocol *Distributed Coordination Function (DCF)* which is a CSMA/CA MAC protocol, and *Point Coordination Function (PCF)* where a point coordinator determines which mobile node is given the right to transmit.

When a mobile node enters in a new BSS, after an idle mode or after moving, it needs to synchronize itself with the access point. To do so, the mobile node has two possibilities: the passive scanning where it waits for a signalization frame periodically sent by the access point, or the active scanning where the mobile node sends a *Probe Request* frame to solicit a *Probe Response* frame. Once the mobile node is synchronized with the access point, it enters into an authentication procedure. If the authentication is successful, the mobile node starts an association process where the access point informs the mobile node about the transmission parameters in the BSS (e.g., the data rate and the transmission power). Once the association completes, the mobile node can communicate via the new access point.

When cover areas of different access points share a common cover zone, the mobile node can handover between the access points. A mobile node associates itself with the access point which offers the best signal or which has the minimum load among the access points. A L2 (Layer 2) handover in IEEE 802.11b WLAN takes place when a mobile node changes its association from one AP (pAP : current AP) to another (nAP : new AP). This process consists of the following steps[6]:

1. The MN performs a scan to see what APs are available. The result of the scan is a list of APs together with physical layer information, such as signal strength.
2. The MN chooses one of the APs and performs a join to synchronize its physical and MAC layer timing parameters with the selected AP.
3. The MN request authentication with the nAP.
4. The MN requests association or re-association with the nAP. A re-association request contains the MAC-layer address of the pAP, while a plain association request does not.
5. If operating in accordance with the IAPP, the nAP performs a lookup based on MAC-layer address to obtain the IP address of the pAP by consulting a local table.

The handover procedure for FMIPv6 in IEEE 802.11b WLAN is shown in Fig. 1. In this figure, the MN performs a discovery of the neighboring ARs after scan. Then the MN may send FBU to the pAR. The pAR receiving the FBU message exchanges HI/Hack messages between the pAR and the nAR and then send FBU message to the MN. At the same time, after the MN sends the FBU message to the pAR, the MN performs join, authentication, re-association, IAPP and then receives FBack message through nAR and nAP.

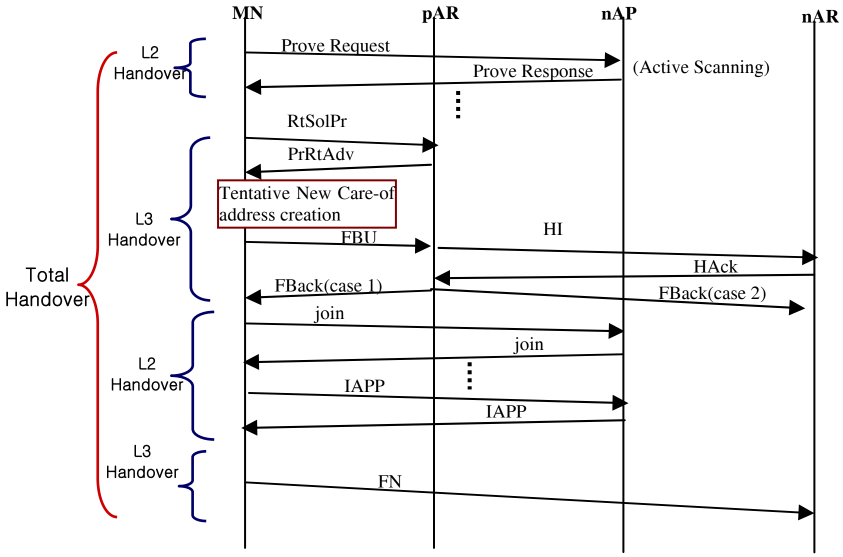


Fig. 1. FMIPv6 handover message flows in IEEE 802.11b WLAN

3 The Proposed Edge-Based Fast Handover

A lot of researches [7]-[13] have tried to reduce the handover latency in IEEE 802.11 networks, but it is not quite improved to reduce the latency. Almost all these results introduce unacceptable delay for real time applications.

The handover latency should be more reduced. Therefore, in this section, new edge-based fast handover mechanism is proposed for the IEEE 802.11b WLAN with several access routers (ARs) connected by access points (APs).

If two or more APs are overlapped and L3 handover between or among the APs is needed, each AP is called *edge-AP* in this research. An AR connected to the edge-AP is called *edge-AR* in this research.

The proposed mechanism uses the network information in local AR's *Candidate Access Router (CAR)* table [3] to predict the handover. In order to perform the mechanism, current edge-AP requests the information of neighboring edge-AP to its local AR using newly defined ICMP messages. The information, called as *edge-based subnet info*, contains neighboring edge-AR's IP addresses, prefix lengths, identities and neighboring edge-AP's BSSIDs.

In order to notify the neighboring edge based subnet info to the MN, new specific subfield is defined and this field is added to beacon message by using the existing reserved field. In this subfield, the neighboring edge-based subnet info achieved from local AR’s CAR table is included. Using these processes, the MN can acquire neighboring network information from beacon message before handover. Then the MN makes nCoAs by using the information and writes it in its own cache together with BSSIDs (AP-IDs) as a searching key. The entry in cache will be used to decide what nCoA is available after scanning.

After above procedures complete, the MN becomes to have the proposed nCoAs and neighboring edge-AP ID in its own cache. There are two possible ways to confirm MN's proposed nCoAs. Firstly, there is no need to confirm because the exchanging new ICMP messages between AP and AR are equal without the confirmation. Therefore, the proposed mechanism can eliminate the procedure of exchanging RtSolPr/PrRtAdv messages between the MN and the AR from FMIPv6 procedure. Secondly, the MN exchanges the messages to confirm before handover.

After these procedures, the MN performs the scan and chooses one of the APs. The chosen AP-ID is compared with each entry, and then if the entry matches the input key (i.e., chosen AP-ID), the searching result is one of nCoAs. Then the MN configures the nCoA and sends FBU message to the pAR. The pAR receiving the FBU message exchanges HI/Hack messages between the pAR and the nAR, and sends FBack message to the MN. At the same time, after the MN sends the FBU message to the pAR, the MN performs join, authentication, re-association, IAPP and then receives FBack message through nAR and nAP. This paper considers this procedure as case 1. The other case (case 2) is that the MN performs join, authentication, re-association, IAPP, after the MN receives FBack message from pAR.

The simulation results show that the handover latency of the proposed mechanism compared with the handover latency of FMIPv6 for these two cases.

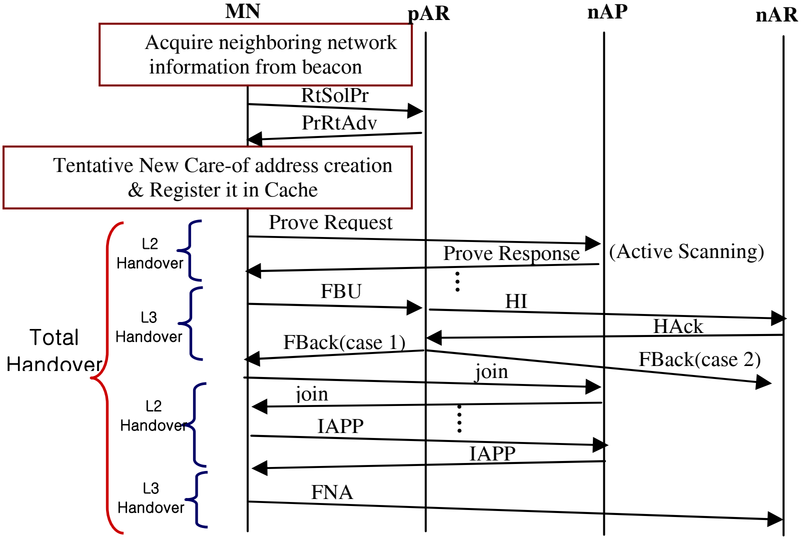


Fig. 2. The proposed edge-based fast handover message flow

In the proposed edge-based fast handover, total handover latency can be more reduced because this approach performs the RtSolPr/PrRtAdv message exchange between pAR and nAR before the MN moves to new AP or can be eliminated the messages which are needed in the FMIPv6. Thus, RtSolPr/PrRtAdv exchange latency is not included in total handover latency and IP configuration latency is reduced by using the cache memory. The message flow of the proposed mechanism is represented in Fig. 2.

4 Simulation

This section gives an overview of the simulation environment. The FMIPv6 and the proposed approach have been implemented using Linux machine and the OMNET++ simulator[14][15].

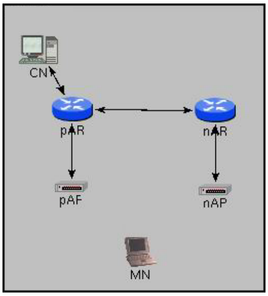
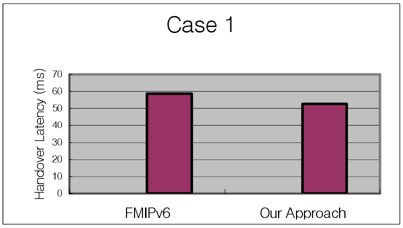


Fig. 3. Simulation Model

The configuration of the simulation network is seen in Fig. 3. Each AR connects to an AP forming an IEEE 802.11b WLAN. An MN roams across different subnets. A CN connecting to one of the AR acts as a data sender for the MN. This demonstrates connectivity between the MN and the CN while the MN changes points of attachments.



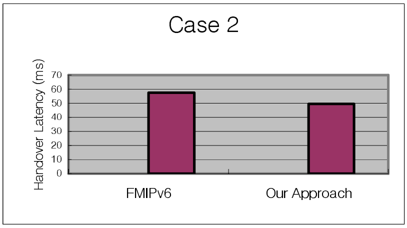


Fig. 4. Simulation results of the case 1 **Fig. 5.** Simulation results of the case 2

In order to make clear comparison, 10 handovers are performed. Averages of handover latency of the results are shown in Fig. 4 and Fig. 5 that the proposed mechanism outperforms FMIPv6. The two cases are considered in the evaluation. The first one is the case(case 1) when the MN receives FBack message from pAR. The comparing results of the case 1 are shown in Fig. 4. The other one is the case (case 2) when the MN receives FBack message through nAR and nAP. The comparing results of the case 2 are shown in Fig. 5.

5 Conclusions

The proposed edge-based fast handover provides several advantages over the ordinary FMIPv6. The first one is handover latency could be reduced by eliminating the router discovery messages (RtSolPr/PrRtAdv) or by exchanging the two messages in advance. The second one is amount of signal loads might be reduced during the handover. The third advantage is the mobile nodes could choose one of the APs flexibly and more efficiently by using the cache memory.

However, it needs additional hardware for performing the proposed approach. Nevertheless it is not a serious problem due to the low cost hardware and the improved technologies to fabricate the required cache memory.

The primary contribution of this research is to provide new approach for reducing the handover latency for fast handover.

References

1. Johnson, D. B., Perkins, C. E., Arkko, J: Mobility Support in IPv6. IETF RFC 3775 (2004)
2. Koodli, R.: Fast Handover for Mobile IPv6. IETF Draft: draft-ietf-mipshop-fast-mipv6-02.txt(2004)
3. Marco Liebsch et al.: Candidate Access Router Discovery. IETF Draft: draft-ietf-seamoby-card-protocol-05.txt(2003)
4. IEEE. Part 11: Wireless LAN Medium Access Control (MAC) and Physical layer (PHY) Specifications. IEEE Standard 802.11 (1999)
5. H. Soliman, C. Castelluccia, K. El-Malki, and L. Bellier.: Hierarchical MIPv6 mobility management (HMIPv6). IETF Draft: draft-ietf-mipshop-hmipv6-00.txt(2003)
6. Yong-Geun Hong, Myung-Ki Shin, Hyoung-Jun Kim, Woo-Suck Cha, Gi-Hwan Cho.: Considerations of FMIPv6 in 802.11 networks. IETF Draft: draft-hong-mobileip-applicability-00.txt(2003)
7. Arunesh Mishra, Minho Shin, William Arbaugh: An Empirical Analysis of the IEEE 802.11 MAC Layer Handoff Process. ACM SIGMOBILE Computer Communication Rev, Vol. 33. (2003) 5-19
8. R. Chakravorty et al.: Mobicom Poster:Practical Experience with Wireless Networks Integration using Mobile IPv6. ACM Mobile Computing and Communicaions Review, Vol. 7. (2003) 47-49
9. Sang-Dong Jang, Wu Woan Kim.: Mobility Support Algorithm Based on Wireless 802.11b LAN for Fast Handover. Lecture Notes in Computer Science, Vol. 3220. Springer-Verlag, Berlin Heidelberg Singapore (2004)
10. R. Chakravorty et al.: On Inter-network Handover Performance using Mobile IPv6. UCCL-Technical Report (2003)
11. Héctor Velayos, Gunnar Karlsson, "Techniques to Reduce IEEE 802.11b MAC Layer Handover Time," KTH, Royal Institute of Technology Technical Report, ISRN KTH/IMIT/LCN/R-03/02 (2003)
12. Nicolas Montavont, Thomas Noël: Analysis and Evaluation of Mobile IPv6 Handovers over Wireless LAN, ACM Mobile Networks and Applications, Vol. 8. (2003) 643-653

13. Xavier Pérez-Costa, Marc Torrent-Moreno, Hannes Hartenstein.: A Performance Comparison of Mobile IPv6, Hierarchical Mobile IPv6, Fast Handovers for Mobile IPv6 and their Combination, ACM SIGMOBILE Mobile Computing and Communication Rev, Vol. 7. (2003) 5-19
14. E. Wu, J. Lai, A. Sekercioğlu.: An Accurate Simulation Model for Mobile IPv6 Protocol. Monash University, Australia, Dept. of Electrical and Computer Systems Engineering (2004)
15. OMNET ++ simulator. http://www.omnetpp.org

An Extendible Hashing Based Recovery Method in a Shared-Nothing Spatial Database Cluster*

Yong-Il Jang[1], Ho-Seok Kim[1], Soon-Young Park[1],
Jae-Dong Lee[2], and Hae-Young Bae[1]

[1] Dept. of Computer Science and Information Engineering, INHA University,
253 Yonghyun-dong, Nam-gu, Incheon, 402-751, Korea
[2] Dept. of Information and Computer Science, DANKOOK University
{yijang, hskim, sunny}@dblab.inha.ac.kr,
letsdoit@dku.edu, hybae@inha.ac.kr

Abstract. In this paper, a recovery method using extendible hashing in a shared-nothing spatial database cluster is proposed. The purpose is to increase the recovery performance and to decrease overhead of the system. In the case of failure, the recovery method in a database cluster restores the database using replicated data from neighbor node. When detect a failure, neighbor node writes the cluster log, and it must be transferred to a failure node. However, in neighbor node, one transaction makes several logs, and increase transferring log size. Also, this increases the recovery time and overhead of the internal network. The proposed method defines a novel cluster log that is composed of update type and a pointer to a record through RID or primary key. This is managed by extendible hashing in main memory. The last transaction replaces the cluster log. Through sending of last updated data, the number of cluster logs and transaction count in failure node are decreased. As a result, the method in this paper increased the availability of the database cluster.

1 Introduction

The recovery method is the most important for high availability in a shared-nothing spatial database cluster. For a good recovery method, many kinds of researches were done for decrease of cluster log's transmission overhead and recovery time in failure node [11].

Each node of shared-nothing spatial database cluster senses failure through the connection state. In the case of cluster node failure, recovery method has two steps. First step is a node recovery like a standalone database management system, and second is a cluster recovery for synchronization with cluster node. In first step, local logs are used for undoing and redoing of local transaction. In second step, the cluster log is used for cluster recovery to synchronize a database with other cluster nodes [1, 8, 10].

* This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment).

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1126 – 1135, 2006.

However, in neighbor node, one transaction makes several cluster logs which increases transferring log size and includes many unnecessary past transactions for one record. Also, unnecessary transactions will be processed in the failure node. The size of spatial transaction related cluster logs is bigger than non-spatial cluster logs. Because cluster recovery is performed in real time, it affects performance of shared-nothing spatial database cluster. Furthermore, it increases the recovery time and overhead of the internal network [5, 7].

In this paper, an extendible hashing based recovery method using the cluster log in a shared-nothing spatial database cluster is proposed. The extendible hashing manages a cluster log in main memory. Each cluster log includes updated information and the pointer of target records by RID or primary keys. As a kind of queries, there are three types of cluster logs which are insert, update, and delete operations. And, In the case of insert and update operation, a RID of target record is wrote in cluster log. In the case of delete operation, because the record had been deleted, instead of RID, a primary key is used.

The proposed method improves abovementioned problems. Through sending of last updated data without overlapping, the number of cluster logs and transaction count in failure node are decreased. First, the small size of data transmission decreased the network overhead. In a shared-nothing spatial database cluster, the network bandwidth is a critical consideration point, because if recovery processing increases network overhead, the whole transaction processing capacity is reduced. Second, the recovery time is reduced by a small transaction processing. As a result, the method in this paper increased the availability of a shared-nothing spatial database cluster.

Contents organization of this paper is as following. Related work about shared-nothing spatial database cluster and its recovery method is presented in Section 2. Section 3 presents the proposed recovery method about hashing structure and its context for recovery sequence. In Section 4 and 5, performance evaluation and conclusion are presented.

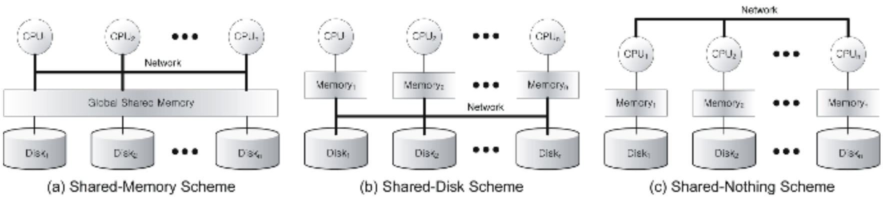
2 Related Work

In this section, firstly shared-nothing database cluster is explained with architecture, data fragmentation policy and replication policy. Second, recovery method for database cluster is explained.

2.1 Database Cluster

A database cluster is a database in which nodes independently capable of providing services are connected to each other through a high speed network and act as a single system [3].

The database cluster provides a division policy, so that a piece of data is divided into small pieces of data and the small pieces of data are managed by different nodes, thus providing high performance to improve simultaneous throughput with respect to an update operation. Further, the database cluster provides a replication policy, so that the duplicates of respective data remain in other nodes, thus providing availability to continuously provide service even if a failure occurs in one node. Such a database cluster includes a shared memory scheme, a shared disk scheme and a shared-nothing scheme, which are shown in Fig. 1 [4].



(a) Shared-Memory Scheme (b) Shared-Disk Scheme (c) Shared-Nothing Scheme

Fig. 1. Schemes of Database Cluster

The shared memory scheme of Fig. 1 (a) is disadvantageous in that a network load excessively increases in order to access the shared memory and in that all processes use the shared memory, so that the disturbance of access to shared resources is increased. Therefore, each node must independently set the size of its cache memory to the maximum.

The shared disk scheme of Fig. 1 (b) is disadvantageous in that, since all nodes share disks, lock frequently occurs with respect to desired resources, and update operations must be equally performed on all disks. Therefore, as the number of disks increases, the load of update operations increases.

The shared-nothing scheme of Fig. 1 (c) is advantageous in that, since the dependence of each node on resources is minimized and each node is not influenced by other nodes. Therefore, it is preferable that the database cluster use the shared-nothing scheme that can be easily extended and has excellent parallelism [1, 4, 8].

2.2 Recovery Method for a Database Cluster

Generally, the recovery of the shared-nothing database cluster includes a node recovery procedure of recovering an individual node and a cluster recovery procedure of recovering cluster configuration.

Node recovery is a recovery procedure of maintaining the consistency of data belonging to a node up to the time when a failure occurs in the node. Cluster recovery is a recovery procedure of maintaining the consistency of data from the time at which the node recovery terminates to the time at which the data participate in the configuration of a cluster when a failure occurs in the node.

If a failure occurs in a node, node recovery is performed to maintain the consistency of the node itself. Thereafter, the recovery of cluster configuration is performed, so that the consistency of operations processed after the failure occurred is maintained. The recovery of cluster configuration is completed, so that the failed node resumes normal service with respect to all operations. ClustRa has a structure in which nodes independently capable of processing queries are connected to each other through a high speed network, and a master node and a backup node form a single group and maintain the same data duplicate. Further, ClustRa maintains the same data duplicate in respective groups using a replication policy applied to groups [2, 3].

If a failure occurs in a node, ClustRa performs a recovery procedure using an internal log required to recover the node itself and distribution logs required to recover cluster configuration. The distribution logs are generated to propagate duplicates in a typical query process and must be stored in a stable storage device. The synchronization of distribution logs is controlled in the duplicates by means of the sequence of logs. However, the recovery technique of ClustRa has the following problem. That is, since node-based distribution logs are maintained in a single queue, the maintenance load for distribution logs is increased, and since the distribution logs are sequentially transmitted to a recovery node, recovery time is increased [5, 11].

GMS/Cluster is a system which has nodes independently capable of processing queries in a shared-nothing structure, and in which 2 to 4 nodes are bundled into a group. The GMS/Cluster uses a complete replication technique allowing all nodes in a group to maintain the same data. If a failure occurs in a node, the GMS/Cluster performs a recovery procedure using a local log required to recover that node and cluster logs required to recover cluster configuration. The local log is equal to a conventional single database log, which must exist in all nodes. The cluster logs are implemented to independently record table-based cluster logs in a master table. If the failed node completes recovery of itself, the node requests cluster logs from other nodes in the group and performs a recovery procedure on the basis of the cluster logs [8, 9, 11].

However, the GMS/Cluster system is problematic in that, since a plurality of pieces of update information are maintained in cluster logs with respect to a single record if a plurality of operations occurs with respect to the single record, the size of the cluster logs increases and transmission cost increases, and since a recovery node repeatedly performs operations several times with respect to a single record, recovery time increases.

3 A Recovery Method Using Extendible Hashing

In this section, we describe extendible hashing based recovery method. Also, managing the cluster log scheme in main memory, log transmission and update scheme, and record refresh scheme is mentioned.

3.1 Cluster Log Management Using Extendible Hashing in Main Memory

Cluster logs are required to recover cluster configuration in a database cluster and are generated separately from local logs required to recover individual nodes. If a failure occurs in a node, other nodes in a group generate cluster logs. Each node independently records cluster logs in a master table corresponding to the node.

Further, a duplicate table of another node having a duplicate of a master table existing in the failed node is selected as a temporary master table. The temporary master table functions as a master table until the failed node completes recovery. A recovery node denotes a node that has failed, receives cluster logs from other nodes in the group and performs a recovery procedure.

Cluster logs are recorded in main memory on the basis of extendible hashing, and are each composed of the update information of a record and a pointer indicating actual data, that is, a Record ID (RID) or primary key information which is one of fields having unique values for each record in the table. If a plurality of operations occurs with respect to a single record after a failure occurs in a node, only the latest update information is maintained in cluster logs using extendible hashing. If an insert operation and an update operation occur, cluster logs are configured on the basis of RID indicating the physical address of data. If a delete operation occurs, cluster logs are configured on the basis of a primary key to identify data. Therefore, the size of maintained cluster logs and information stored therein vary according to the type of operations that occurred [6].

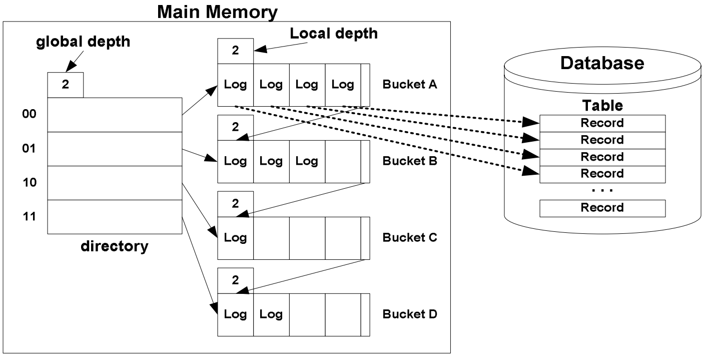


Fig. 2. An Extendible Hashing-Based Cluster Log Management Structure

Fig. 2 is an extendible hashing-based cluster log management structure. Data required to manage cluster logs maintained in the main memory are composed of a global depth, a local depth, a directory and buckets. Information about each element is described below.

A global depth is an index for a directory which denotes the size of a current directory, and a local depth denotes the occurrence of overflow from a corresponding bucket. A directory stores a pointer indicating buckets. Each bucket stores cluster logs maintaining the latest update information. Each bucket supports combination and division according to a cluster log, and the directory supports only division, so that a structure of decreasing operation cost in the main memory is implemented.

Further, each bucket sequentially accesses cluster logs in a connection list structure to flexibly configure packets at the time of transmitting the cluster logs to a recovery node. Each of cluster logs stored using extendible hashing is composed of the update information of a record and information indicating actual data stored in a database. The database maintains actual data on which an operation occurs.

3.2 Structure of the Cluster Log

Fig. 3 is the configuration of the cluster log of Fig. 2. If an insert operation occurs, data are inserted in a master table and an index is generated on the basis of RID of the data to be inserted. Further, an I flag indicating that the insert operation has occurred, and an RID which is the physical address of actual data stored in the database remain in a cluster log.

Recorded Log	Transmission Data	
INSERT	I	Data

Recorded Log	Transmission Data		
UPDATE	U	Data	old PK

Recorded Log	Transmission Data	
DELETE	D	Data

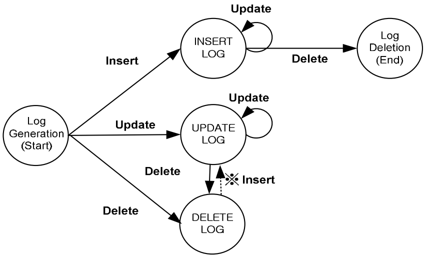
I : Insert **U : Update**
D : Delete PK : Primary Key

Fig. 3. Configuration of the Cluster Log

If an update operation occurs, data are updated in the master table and an index is updated on the basis of RID of the data to be updated. Further, a U flag indicating that the update operation has occurred, an RID which is the physical address of actual data stored in the database, and a primary key of old data (old primary key: old PK) which is to be updated, remain in a cluster log. If a delete operation occurs, data are deleted from the master table, and an index is updated on the basis of a primary key of data to be deleted. Further, a D flag indicating that the delete operation has occurred, and a primary key (primary key: PK) of data to be deleted remain in a cluster log.

3.3 Writing Logs of the Updated Record

The recording of cluster logs is performed so that a directory address is searched for using results, obtained by applying RID of a corresponding record to a hash function, in binary notation, and the cluster logs are recorded in a bucket indicated by a pointer stored in a directory.



(a) A Process of Allowing a Cluster Log According to Operations

	Previously Recorded Cluster Log			
New Log	-	Insert Log	Update Log	Delete Log
Insert Log	Insert Log			
Update Log	Update Log	Insert Log	Update Log	
Delete Log	Delete Log	Log Deletion	Delete Log	

(b) A Table of Latest Cluster Log According to Previous Log

Fig. 4. The Recording of Cluster Logs

The recording of cluster logs is performed so that a master node independently records logs according to a generated transaction when a failure occurs in a node. Further, if a plurality of operations occurs before cluster logs are reflected on a recovery node, the operations are processed in a corresponding master table, and then the latest update information is reflected in the cluster logs. This operation allows only the latest update information to remain in the cluster logs even if a plurality of operations has occurred, so that the recovery node performs a single operation, thus maintaining consistency with other nodes. Further, this operation causes the size of cluster logs to decrease, thus supporting the rapid recovery of the recovery node.

Fig. 4 is the recording of cluster logs maintaining the latest update information. (a) illustrates a process of allowing a cluster log to maintain the latest update information according to operations occurring after the cluster log was generated, and (b) illustrates that only the latest cluster log is maintained, even if a plurality of operations has occurred to perform a recovery operation. In Fig. 4, "※" indicates that, when an insert log is generated after a delete log is recorded, primary keys of the delete and insert logs are compared to each other, and an update log or insert log is recorded.

After a cluster log is generated, an insert operation, an update operation and a delete operation can occur. "Start" denotes the generation of the cluster log. If new data are inserted in a master table after the occurrence of a failure, an insert log is recorded to apply the new data to a recovery node. If old data are updated, an update log is recorded to reflect the updated data. If data are deleted, a delete log is recorded. If the recorded log is transmitted to the recovery node and reflected on the recovery node, a corresponding cluster log is deleted. If a plurality of operations occurs before the recorded log is transmitted to the recovery node, the latest update information remains in the cluster log.

3.4 The Transmission of Cluster Logs

Cluster recovery is performed by receiving cluster logs, so that the transmission of cluster logs greatly influences cluster recovery.

In the meantime, a recovery node performs node recovery at the first step. Node recovery uses a local log left when an update operation has occurred on the data of a node. If node recovery is completed, the recovery node maintains the consistency of data thereof, and performs cluster recovery that is the second step required to maintain the consistency of cluster configuration. At the time of cluster recovery, the recovery node informs other nodes that the recovery node has recovered, and requests recorded cluster logs from the other nodes. The other nodes sense that the recovery node has completed node recovery at the first step, and transmit the cluster logs to the recovery node in packets.

In Fig. 5, a processing position moves to a first bucket to transmit cluster logs at step ①, and a packet is initialized at step ②. Cluster logs that are stored in each bucket and maintain the latest update information are sequentially accessed using a connection list. Actual data are duplicated in and added to a packet on the basis of RID stored in the cluster logs at step ③. If a single packet is configured, the packet is transmitted to the recovery node at step ④.

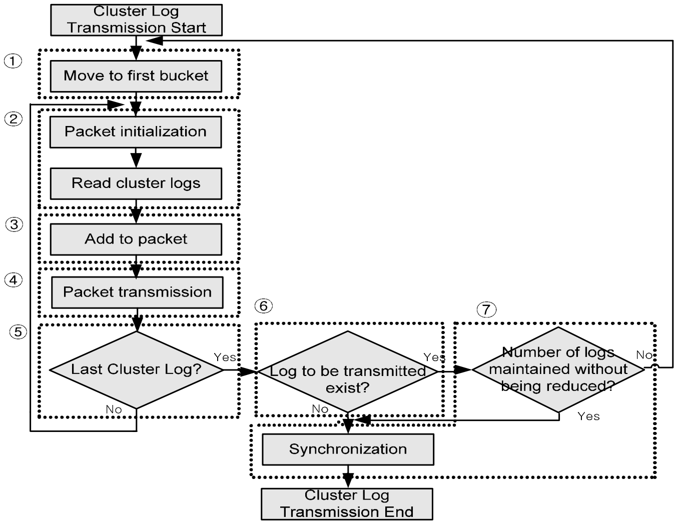


Fig. 5. A Flowchart for a Process of Transmitting Cluster Logs

An initialization procedure is performed with respect to a packet for which an acknowledgement is received and a bucket is accessed to configure cluster logs as a packet until the last cluster log has been transmitted to the recovery node at step ⑤. If a transaction occurs during transmission, the transaction is processed by a corresponding master table, so a cluster log remains. Each bucket is examined to determine whether a cluster log to be transmitted exists at step ⑥. If no cluster log to be transmitted exists, a synchronization procedure is performed.

If a transaction occurs during the transmission and cluster logs exist at step ⑥, the processing position returns to the first bucket, and the remaining cluster logs are configured as a packet and transmitted to the recovery node. If cluster logs exist during transmission, but the number of cluster logs is maintained at a certain number without decreasing at step ⑦, a synchronization procedure is compulsorily performed.

The synchronization procedure is a procedure of consistently maintaining all cluster logs. That is, the master table is changed to a temporary standby state, so that all transactions occurring in the standby state stand by in the queue of the master table. If the last cluster log packet is reflected on the recovery node, the transactions standing by in the queue are transmitted to the recovery node, thus maintaining consistency with other nodes. The recovery node returns to a state existing prior to the occurrence of failure, thus terminating the reflection of cluster logs.

4 Performance Evaluation

In this section, we will present the result of experiments to analyze the performance of the proposed method with respect to the number of logs and the update query processing performance. For comparing and verifying the effectiveness of the proposed method, GMS/Cluster is used on Windows XP PC with Pentium 4 2.4G Hz CPU, 1

GB memory and 80 GB HDD. We assumed that 10 tables are stored in database with one master node and three backup nodes.

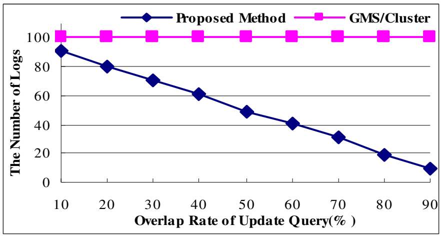


Fig. 6. Search Performance

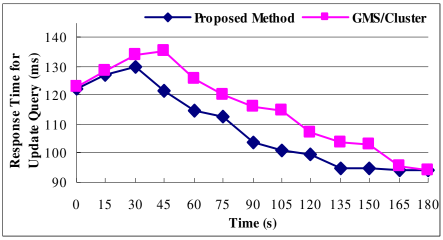


Fig. 7. Update Performance

In the first experiment, we compared the number of logs of GMS/Cluster and proposed recovery method. As the overlapped updated query rate grows, in proposed method, the number of logs decreased than GMS/Cluster's recovery. Second experimental point is focused on update query processing time during recovery. With 50% of overlapped update query rate, the response time has been slightly decreased. Therefore transaction throughput is increased. Also, the recovery term is decreased. Completion of recovery term is shorter than 45 second.

5 Conclusion

The proposed method provides a recovery method using extendible hashing-based cluster logs in a shared-nothing spatial database cluster. It maintains only the update information of a record and RID or primary key as a pointer indicating actual data in main memory and stores the latest update information in cluster logs when a plurality of operations occur with respect to a single record.

Therefore, the number of cluster logs decreases, and a transmission load decreases when the cluster logs are transmitted to a recovery node. Further, the proposed method is advantageous in that a recovery node need only perform a single update operation with respect to a single record, thus decreasing recovery time.

Further, it is advantageous in that, since it manages cluster logs using main memory-based extendible hashing, the maintenance load for cluster logs decreases, and since a load attributable to a node failure is decreased in a recovery node, recovery time decreases, so that stable service can be continuously provided, thus consequently improving the performance of a shared-nothing spatial database cluster.

References

1. R. Bamford, Rafiul Ahad, Angelo Pruscino, A Scalable and Highly Available Networked Database Architecture, In Proc. of the 25th VLDB Conference, Edinburgh, Scotland, 1999.
2. Philip A. Bernstein, Nathan Goodman, The Failure and Recovery Problem for Replicated Databases, Second ACM Symposium on the Principles of Distributed Computing, 1983.

3. Svein Erik Bratsberg, Svein-Olaf Hvasshovd, Oystein Torbjornsen, Parallel Solutions in ClustRa, IEEE Computer Society Technical Committe on Data Engineering, 1997.
4. David J. DeWitt, Jim Gray, Parallel Database Systems: The Future of Database Processing or a Passing Fad?, ACM SIGMOD Record, Special Issue on Directions for Future Database Research and Development.
5. Svein-Olaf Hvasshovd, Oystein Torbjornsen, Svein Erik Bratsberg, The ClustRa Telecom Database: High Availability, High Throughput, and Real-Time Response, In Proc. of the 21st VLDB Conference, 1995.
6. Ronald Fagin, Jurg Nievergelt, Nicholas Pippenger, H. Raymond Strong, Extendible hashing: A Fast Access Method for Dynamic Files, ACM Transactions on Database Systems, 1979.
7. R. Jimenez-Peris, M. Patino-Martinez, G. Alonso, An Algorithm for Non-Intrusive, Parallel Recovery of Replicated Data and its Correctness, IEEE Symp. on Reliable Distributed Systems, 2002.
8. Chung-Ho Lee, A Partial Replication Protocol and a Dynamically Scaling Method for Database Cluster Systems, PhD thesis, Department of Computer Science & Information Engineering, Inha Univ., Korea, 2003.
9. Byeong-Sub Ryu, The Collaborative Cluster Recovery Method of a Shared-Nothing Spatial Database Cluster, Master thesis, Department of Computer Science & Information Engineering, Inha Univ., Korea, 2004.
10. Maitrayi Sabaratnam, Oystein Torbjornsen, Evaluating the Effectiveness of Fault Tolerance in Replicated Database Management Systems, Twenty-Ninth Annual International Symposium on Fault-Tolerant Computing, 1999.
11. Oystein Torbjornsen, Svein-Olaf Hvasshovd, Young-Kuk Kim, Towards Real-Time Performance in a Scalable, Continuously Available Telecom DBMS, ClustRa, 2001.

A Quantitative Justification to Partial Replication of Web Contents*

Jose Daniel Garcia[1], Jesus Carretero[2], Felix Garcia[2], Javier Fernandez[2], Alejandro Calderon[2], and David E. Singh[2]

[1] Universidad Carlos III de Madrid, Computer Architecture Group,
Avda. de la Universidad Carlos III, 22. 28270 Colmenarejo, Madrid, Spain
[2] Universidad Carlos III de Madrid, Computer Architecture Group,
Avda de la Universidad 30. 28911 Leganés, Madrid, Spain

Abstract. Traditionally the alternatives for Web content storage have been full replication and full distribution. More recently partial replication has been proposed as an hybrid strategy. This paper shows a quantitative justification to advantages achieved by using this approach in terms of storage capacity usage and reliability. Our analytical study proves that partial replication offers much higher storage capacity than full replication and that its reliability is much higher than full distribution reaching to levels equivalent to those provided by full replication.

1 Introduction

In the last years the demand of high performance internet Web servers has increased dramatically. The target is a solution which is scalable in both performance and storage capacity while reliability is not compromised. The traditional standalone Web server approach presents severe scalability limits which may be partly mitigated by means of hardware scale-up [1] which may be viewed as a short term solution. Web server performance may also be improved by acting on the operating system [2, 3, 4, 5] or on the Web server software itself [6, 7].

Another approach is the distributed Web server [8] where a set of server nodes are used to host a Web site. In these systems performance scalability may be achieved by adding new server nodes to the system. Three approaches have been used to allocate contents to server nodes.

Full replication. In a fully replicated solution every file is replicated in every server node [9, 1]. This solution provides a high reliability, as any file may be served by any node, although there are limitations on the storage capacity which is limited by the server node with the lowest capacity.

* This work has been partially supported by the Spanish Ministry of Science and Education under the TIN2004-02156 contract.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1136–1145, 2006.

Full distribution. In a fully distributed solution each file is allocated in a single server node [10]. This solution is not limited in storage capacity although it is not fault tolerant at all. If there is a fault in a server node a subset of the contents is not available.

Partial replication. In a partially replicated solution each file is stored in a subset of server nodes [11, 12, 13, 14]. This solution provides better reliability than the fully replicated one and higher storage capacity than the fully distributed one (depending on the replication degree of each file).

Different architectures have been proposed for distributed Web servers. Solutions range from *Distributed Web systems* [15] and *Virtual Web Clusters* [16] to *Cluster Based Web Systems* [17, 18, 19] (also known as *Web Clusters*). In all of them a set of server nodes offer a single system image providing service to Web requests. Solutions vary in their topology and in the method used to route request to server nodes.

While *Web Clusters* and *Distributed Web Systems* may support partial replication of contents, it is not easy to integrate a partial replication strategy into a virtual Web cluster [20]. In practice, partial replication solutions have only been proposed for *Web clusters*. In this paper we study the storage capacity and reliability of partially replicated solutions and we compare to fully replicated and fully distributed solutions.

2 Storage Capacity

In the following discussion the website is represented by a set E of N elements e_i, where each element has size s_i. A distributed web server is represented by the set X of M server nodes x_j, where each server node has a storage capacity c_j. Given this, the size of the full website S is represented by (1).

$$S = \sum_{i=1}^{N} s_i \tag{1}$$

In general, the allocation of elements to server nodes may be represented by the allocation matrix A, where a_{ij} takes value 1 if element e_i is allocated on server node x_j, and takes value 0 otherwise. Given this allocation matrix, the occupancy V_j of a server node x_j is expressed by (2) and the global occupancy of the website in a distributed Web server is given by (3).

$$V_j = \sum_{i=1}^{N} a_{ij} s_i \tag{2}$$

$$V = \sum_{j=1}^{M} V_j = \sum_{j=1}^{M} \sum_{i=1}^{N} a_{ij} s_i \tag{3}$$

Contents stored in each server node must fit in the storage capacity of that node as expressed in (4).

$$V_j \leq c_j \; \forall j \in [1, M] \tag{4}$$

2.1 Full Replication

With full replication every element is allocated to every server node. That is, $a_{ij} = 1 \; \forall i \in [1, N] \; \forall j \in [1, M]$. This gives restrictions expressed in (5).

$$\sum_{i=1}^{N} a_{ij} = N \; \sum_{j=1}^{M} a_{ij} = M \tag{5}$$

Combination of (3) and (5) yields (6). That is, the global occupancy in a fully replicated distributed Web server is the number of server nodes times the size of the website.

$$V = \sum_{j=1}^{M} \sum_{i=1}^{N} a_{ij} s_i = \sum_{i=1}^{N} \sum_{j=1}^{M} a_{ij} s_i = \sum_{i=1}^{N} s_i \sum_{j=1}^{M} a_{ij} = \sum_{i=1}^{N} s_i M = M \cdot S \tag{6}$$

In a fully replicated distributed Web server, every file is stored in every server node and, consequently, $V_j = S \; \forall j in \, [1, N]$. In addition, the size of the website must satisfy the restriction that it is below the storage capacity of each server node as stated in (7), which leads to the general restriction that the size of the website is limited by the server node with the lowest storage capacity.

$$S \leq c_j \; \forall j \in [1, M] \; \Rightarrow \; S \leq \min_{k \in [1,M]} \{c_j\} \tag{7}$$

It is important to note that adding new server nodes to a fully replicated distributed Web server does not scale up its storage capacity. The only way to improve storage capacity, in that case, is to improve the storage capacity of the server node with the lowest capacity. Improving storage capacity of a server node may be performed either by adding a new storage device or by replacing a storage device.

In any case, after adding new storage resources to the Web server the global storage capacity is increased in ΔC (from C to C') and the maximum size of the hosted website is increased in ΔS (from S to S'). To measure the storage capacity obtained when adding new storage resources we define the concept of storage improvement efficiency E as the ratio between size increment of the website and the storage capacity which was necessary to acquire ($E = \Delta S / \Delta C$).

Efficiency of storage device addition. Let x_k be the server with lowest capacity c_k ($c_k = \min_{i \in [1,M]} \{c_i\}$). If server node x_k is added a new storage device of capacity Δc_k, the new capacity for server node x_k is $c'_k = c_k + \Delta c_k$. Now, the size of the website is limited by the server node with the lowest storage capacity, excluding x_k. The increment in the size of the website that can be hosted in the distributed Web server ΔS is given in (8).

$$\Delta S = \min \left\{ \Delta c_k, \min_{\substack{i \in [1,M] \\ i \neq k}} \{c_i - c_k\} \right\} \tag{8}$$

If several server nodes had initially the minimum capacity, no increment is obtained unless all those nodes are added new storage devices of capacity Δc_k each one. If the number of server nodes with minimum storage capacity is initially M_{min} (with $1 \leq M_{min} \leq M$), the global capacity increase may be expressed as $\Delta C = M_{min} \cdot \Delta c_k$. Thus, the efficiency of storage device addition may be expressed by (9).

$$E = \frac{\min\left\{ \Delta c_k, \min\limits_{\substack{i\in[1,M]\\ i\neq k}} \{c_i - c_k\} \right\}}{M_{min} \cdot \Delta c_k} \tag{9}$$

When Δc_k is below the limit imposed by the rest of server nodes, efficiency takes value $1/M_{min}$. When Δc_k increases over that limit, efficiency decreases. This decrease is due to the fact that with high values for Δc_k not all the available space may be used. Furthermore, for very high values of Δc_k efficiency approaches to zero.

In a distributed Web server with homogeneous storage capacities web site size increment is always Δc_k. Thus, in such a case the efficiency is constant, as expressed by $E = \Delta c_k / (M \Delta c_k) = 1/M$.

Efficiency of storage device replacement. If server node x_k cannot be added a new storage device, the other alternative is to replace its current storage device of capacity c_k by a new storage device of capacity $c_k + \Delta c_k$. The increment in the maximum website size is the same than the one determined for storage device addition. However the storage capacity acquirement ΔC now takes the value $c_k + \Delta c_k$ (for one disk). Thus the general expression of the efficiency of storage device replacement may be expressed by (10).

$$E = \frac{\min\left\{ \Delta c_k, \min\limits_{\substack{i\in[1,M]\\ i\neq k}} \{c_i - c_k\} \right\}}{M_{min} \left(c_k + \Delta c_k\right)} \tag{10}$$

When Δc_k is below the limit imposed by the rest of server nodes, efficiency takes values below $1/M_{min}$. When Δc_k increases over that limit, efficiency decreases. This decrease is due to the fact that with high values for Δc_k not all the available space may be used. Furthermore, for very high values of Δc_k efficiency approaches to zero. In a distributed Web server with homogeneous storage capacities web site size increment is always Δc_k. Thus, in such a case efficiency is expressed by (11).

$$E = \frac{\Delta c_k}{M \left(c_k + \Delta c_k\right)} \tag{11}$$

For very high values of Δc_k, efficiency approaches to 1 as we show in (12). That is, in a fully replicated distributed Web server efficiency is always below $1/M$.

$$\lim_{\Delta c_k \to \infty} \frac{\Delta c_k}{M \left(c_k + \Delta c_k\right)} = \lim_{\Delta c_k \to \infty} \frac{1}{\frac{M c_k}{\Delta c_k} + M} = \frac{1}{M} \tag{12}$$

2.2 Full Distribution

With full distribution every element is allocated to one and only one server node. This leads to restriction expressed in (13).

$$\sum_{i=1}^{M} a_{ij} = 1 \; \forall i \in [1, M] \tag{13}$$

Combination of (13) and (3) yields to (14). That is, global storage occupancy is exactly the size of the website.

$$V = \sum_{i=1}^{N} s_i \sum_{j=1}^{M} a_{ij} = \sum_{i=1}^{N} s_i = S \tag{14}$$

Family of restrictions expressed by (4) may be added giving the aggregated restriction expressed by (15).

$$\sum_{k=1}^{M} V_k \leq \sum_{k=1}^{M} c_k \tag{15}$$

Left side of (15) is just the size of the website ($\sum_{k=1}^{M} V_k = S$) as each file is stored in one and only one server node. Thus, (15) may be expressed in the form of (16). That is, the size of the website is limited by the aggregated capacity of all the nodes in the cluster. In this case, adding a new server node to the cluster does scale up its storage capacity. Improving storage capacity of an existing node (either by adding a new storage device or by replacing an existing one) also scales up storage capacity of the cluster.

$$S \leq \sum_{k=1}^{M} c_k \tag{16}$$

Efficiency of storage device addition. When a new storage device is added to an existing node x_j, storage capacity is increased Δc_j (from c_j to $c'_j = c_j + \Delta c_j$) and the maximum size of the hosted website is given by (17).

$$S' \leq \Delta c_j + \sum_{k=1}^{M} c_k \tag{17}$$

Thus, the increment in the size of the website which can be hosted is $\Delta S = S' - S = \Delta c_j$, and the efficiency of storage device addition is $E = \Delta c_j / \Delta c_j = 1$. That is, in a distributed Web server with full distribution of contents storage device addition is always maximum, and all the added storage capacity is used.

Efficiency of storage device replacement. When a storage device of a server node x_j with capacity c_j is replaced by new device of higher storage capacity

$c'_j = c_j + \varDelta c_j$, the maximum size of the hosted website is increased in $\varDelta c_j$. To get this increase it is necessary to acquire a capacity of c'_j. Thus, the efficiency of storage device replacement is given by (18).

$$E = \frac{\varDelta S}{\varDelta C} = \frac{\varDelta c_j}{c_j + \varDelta c_j} \tag{18}$$

The maximum efficiency is reached when $\varDelta c_j$ takes very high values. This maximum value is 1, as we show in (19). Thus, to get a good efficiency in storage device replacement the storage increment must be much higher than old capacity of the server node.

$$\lim_{\varDelta c_j \to \infty} E = \lim_{\varDelta c_j \to \infty} \frac{\varDelta c_j}{c_j + \varDelta c_j} = \lim_{\varDelta c_j \to \infty} \frac{1}{\frac{c_j}{\varDelta c_j} + 1} = 1 \tag{19}$$

2.3 Partial Replication

With partial replication each element is allocated to several server nodes. In this study we consider the particular case in which every file has the same number of replicas r (with $1 < r < M$). That is expressed by (20).

$$\sum_{j=1}^{M} a_{ij} = r \ \forall i \in [1, N] \tag{20}$$

Combination of (3) and (20) yields to (21). That is storage occupancy of a replicated Web site is r times the website size.

$$V = \sum_{j=1}^{M} \sum_{i=1}^{N} a_{ij} s_i = \sum_{i=1}^{N} s_i \sum_{j=1}^{M} a_{ij} = r \sum_{i=1}^{N} s_i = rS \tag{21}$$

As the global occupancy must not exceed the global storage capacity, there is a restriction for the maximum size of the Web site, as shown in(22).

$$rS \leq \sum_{j=1}^{M} c_j \Rightarrow S \leq \frac{1}{r} \sum_{j=1}^{M} c_j \tag{22}$$

Efficiency of storage device addition. When a new storage device is added to an existing node x_j, storage capacity is increased in $\varDelta c_j$ (from c_j to $c'_j = c_j + \varDelta c_j$) and the maximum size of the hosted website is given by (23).

$$S' = \frac{1}{r} \left(\varDelta c_j + \sum_{k=1}^{M} c_k \right) \tag{23}$$

Thus, the increment in the size of the website which can be hosted is $\varDelta S = S' - S = \varDelta c_j / r$, and the efficiency of storage device addition is given by (24). So, efficiency is constant and only dependent on the number of replicas per element, which is a configuration parameter established on system setup.

$$E = \frac{\Delta S}{\Delta C} = \frac{\frac{\Delta c_j}{r}}{\Delta c_j} = \frac{1}{r} \tag{24}$$

Efficiency of storage device replacement. When a storage device of a server node x_j with capacity c_j is replaced by a new device of higher storage capacity $c'_j = c_j + \Delta c_j$, the maximum size of the hosted website is increased in $\Delta c_j / r$, and efficiency of storage replacement is given by (25).

$$E = \frac{\Delta S}{\Delta C} = \frac{\frac{\Delta c_j}{r}}{c_j + \Delta c_j} = \frac{1}{r} \frac{\Delta c_j}{c_j + \Delta c_j} \tag{25}$$

For low increments in storage replacement, efficiency is near to zero, as c_j is dominant in efficiency expression. When Δc_j is high compared with c_j, efficiency increases. Maximum value is obtained when Δc_j is very high compared with c_j as shwon in(26).

$$\lim_{\Delta c_j \to \infty} \frac{1}{r} \frac{\Delta c_j}{c_j + \Delta c_j} = \lim_{\Delta c_j \to \infty} \frac{1}{r} \frac{1}{\frac{c_j}{\Delta c_j} + 1} = \frac{1}{r} \tag{26}$$

3 Reliability

In a cluster based web system there are two main components: the Web switch and the server nodes. The web switch and the set of server nodes form a serial system and its reliability is expressed as $R_{server} = \rho R_{nodes}$, where the reliability of the system (R_{server}) is given in terms of Web switch reliability, ρ, and reliability of the set of server nodes R_{nodes}.

Let M be the number of server nodes and r the number of replicas for each element (assuming all elements have the same number of replicas). Let R_i be the reliability of server node x_i. A failure in the set of nodes happens when a failure arises simultaneously in the r servers where a requested element is stored. The probability of this event is expressed by (27), provided that E_i represents the event that the requested element is stored on server node x_i.

$$F_{nodes} = \sum_{\substack{i_1, i_2, \ldots, i_r = 1 \\ i_1 < i_2 < \ldots < i_r}}^{M} P\left(\bigcup_{j=i_1}^{i_r} E_i \right) \prod_{j=i_1}^{i_r} (1 - R_j) \tag{27}$$

If uniform distribution of replicas among server nodes ($P(E_i) = P(E_j)\ \forall i, j$) is assumed, the probability that the r replicas of an element are in an specific subset of nodes $\{x_{i_1}, x_{i_2}, \ldots, x_{i_r}\}$ is $M! \, (M-r)!/r!$. In addition, we also assume

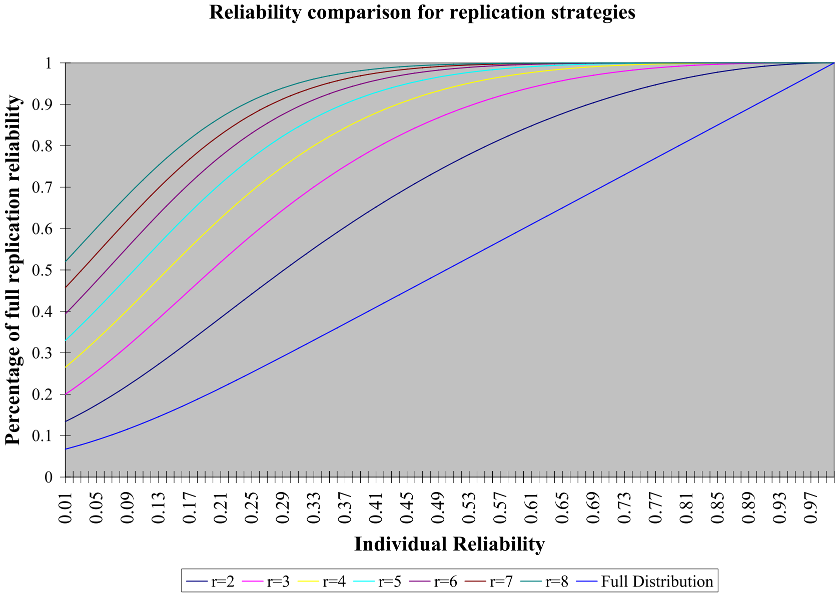


Fig. 1. Reliability percentage of partial replication and full distribution over the corresponding reliability of a fully replicated system

equal reliability among server nodes, which is an acceptable hypothesis when all nodes have the same set of characteristics. With such assumptions, failure probability is shown in (28) for a set of nodes. Consequently, reliability is given by (29).

$$F_{nodes} = \sum_{\substack{i_1, i_2, \ldots, i_r = 1 \\ i_1 < i_2 < \ldots < i_r}}^{M} \frac{M!\,(M-r)!}{r!}\,(1-R)^r = (1-R)^r \tag{28}$$

$$R = \rho\,(1-(1-R)^r) \tag{29}$$

For the particular cases of full replication ($r = M$) and full distribution ($r = 1$) reliability is given by (30) and (31) respectively.

$$R = \rho\left(1-(1-R)^M\right) \tag{30}$$

$$R = \rho R \tag{31}$$

Figure 1 shows the reliability percentage of partial replication and full distribution over the corresponding reliability of a fully replicated system. When each server node has an individual reliability higher than 0.7, a partially replicated solution gets more than 90% of the reliability of a fully replicated system. More importantly, when the partially replicated solution uses at least three replicas per file, the reliability rises up to 99% of the reliability obtained with a fully replicated system. We conclude that partially replicated system, with a low number of replicas per file, may offer a reliability equivalent to the one offered by a fully replicated system but with a lower storage occupancy.

4 Conclusions

In this paper we have studied storage capacity and reliability of partially replicated *Web clusters* compared with fully replicated and fully distributed solutions.

In all cases storage addition should be preferred to storage replacement because the former offers better efficiency than the latter. However, when the storage size increase is high compared with the previous storage capacity of the affected node, efficiency approaches to the case of storage addition. Thus, in that case both solutions (addition and replacement) may be seen as equivalent.

Full distribution is the strategy which offers better efficiency ($E = 1$). And that efficiency is independent of the number of nodes forming the server. On the other hand, full replication offers the lowest efficiency which decreases as the number of server nodes increases. Partial replication strategy offers an intermediate efficiency which only depends on the number of replicas per element. This means that adding new server nodes does not affect the efficiency.From the storage capacity point of view the best strategy is full distribution. However full distribution is the least reliable solution in contrast with full replication which is the most reliable solution. Besides from the reliability point of view a partially replicated system, with a low number of replicas per file, may offer a reliability level equivalent to the one offered by a fully replicated system.

5 Future Work

We are currently working on the removal of equal number of replicas assumption. Special interest is in the case where the number of replicas is based on file popularity and file size. Ongoing work is also in the evaluation of the reliability increase of multiple switched Web clusters.

References

1. Devlin, B., Gray, J., Laing, B., Spix, G.: Scalability terminology: Farms, clones, partitions, and packs: Racs and raps. Technical Report MS-TR-99-85, Microsoft Research. Advanced Technology Division (1999)
2. Banga, G., Druschel, P., Mogul, J.C.: Better operating system features for faster network servers. Performance Evaluation Review **26** (1998) 23–30
3. Banga, G., Druschel, P., Mogul, J.C.: Resource containers: A new facility for resource management in server systems. In: OSDI '99: Proceedings of the third symposium on Operating systems design and implementation. Volume 99. (1999) 45–58
4. Pai, V.S., Druschel, P., Zwaenepoel, W.: Io-lite: A unified i/o buffering and caching system. ACM Transactions on Computer Systems **18** (2000) 37–66

5. Hu, Y., Nanda, A., Yang, Q.: Measurement, analysis and performance improvement of the Apache Web Server. International Journal of Computers and their Applications **8** (2001) 217–231
6. Pai, V.S., Druschel, P., Zwaenepoel, W.: Flash an efficient and portable web server. In: Proceedings of USENIX 1999 Annual Technical Conference. (1999) 199–212
7. Shukla, A., Li, L., Subramanian, A., Ward, P.A.S., Brecht, T.: Evaluating the performance of user-space and kernel-space web servers. In: CASCON '04: Proceedings of the 2004 conference of the Centre for Advanced Studies on Collaborative research, Markham, Ontario, Canada, IBM Press (2004) 189–201
8. Cardellini, V., Casalicchio, E., Colajanni, M., Yu, P.S.: The state of the art in locally distributed Web-server systems. ACM Computing Surveys **34** (2002) 263–311
9. Kwan, T.T., McGrath, R.E., Reed, D.A.: Ncsa's world wide web server: design and performance. IEEE Computer **28** (1995) 68–74
10. Baker, S.M., Moon, B.: Distributed cooperative web servers. Computer Networks **31** (1999) 1215–1229
11. Li, Q., Moon, B.: Distributed cooperative Apache Web server. In: Proceedings of the tenth international conference on World Wide Web, Hong Kong, ACM Press (2001) 555–564
12. Carretero, J., Fernndez, J., Prez, J.M.: Dynamic distribution and allocation of web objects. In: Proceedings of the International Conference on Information Systems, Analysis and Synthesis (ISAS'2001). Volume XII., Orlando, FL, USA (2001)
13. Zhuo, L., Wang, C., Lau., F.C.M.: Document replication and distribution in extensible geographically distributed web servers. Journal of Parallel and Distributed Computing **63** (2003) 927–944
14. Tse, S.S.H.: Approximate algorithms for document placement in distributed web servers. IEEE Transactions on Parallel and Distributed Systems **16** (2005) 489–496
15. Aversa, L., Bestavros, A.: Load balancing of a cluster of web servers: using distributed packet rewriting. In: Conference Proceedings of the 2000 IEEE International Performance, Computing and Communications Conference (IPCCC 2000). (2000) 24–29
16. Vaidya, S., Christensen, K.J.: A single system image server clustering using duplicated mac and ip addresses. In: Proceedings of the 26th Annual IEEE Conference on Local Computer Networks (LCN 2001). (2001) 206–214
17. Dias, D.M., Kish, W., Mukherjee, R., Tewari, R.: A scalable highly available Web server. In: Proceedings of the IEEE International Computer Conference (COMPCON'96), Santa Clara, CA, IEEE (1996) 85–92
18. Aron, M., Druschel, P., Zwaenepoel, W.: Efficient support for P-HTTP in cluster-based web servers. In: Proceedings of the 1999 USENIX Annual Technical Conference, Monterey, CA, USA, USENIX (1999) 185–198
19. Schroeder, T., Goddard, S., Ramamurthy, B.: Scalable web server clustering technologies. IEEE Network **14** (2000) 38–45
20. Garcia, J.D., Carretero, J., Perez, J.M., Garcia, F., Fernandez, J.: A distributed web switch for partially replicated contents. In: 7th World Multiconference on Systemics, Cybernetics and Informatics (WMSCI03). Volume 7. (2003) 1–6

Content Distribution Strategy Using Web-Cached Multicast Technique

Backhyun Kim and Iksoo Kim

Department of Information and Telecommunication Engineering Univ. of Incheon,
177 Namku Towhadong Incheon, Korea
{hidesky24, iskim}@incheon.ac.kr

Abstract. In this paper, we propose content distribution strategy to evenly disperse traffic over network and to reduce the required bandwidth for transmitting content data by merging the adjacent multicasts depending upon the number of proxies ***n*** that have requested the same one. In our technique, streaming for the identical content is fragmented as long as the grouping interval for batching multicast and can be stored among proxies in order of the requests. A client might have to download data on two channels simultaneously, one from server through multicast and the other from proxies through unicast or multicast. According to the popularity of content, the grouping interval of multicast can be dynamically expanded up to ***n*** times and so it can be reduced server's workload and network traffic. We adopt the cache replacement strategy as LFU (Least-Frequently-Used) for popular content, LRU (Least-Recently-Used) for unpopular content, and the method for replacing the first block of content last to reduce end-to-end latency. We perform simulations to compare its performance with that of conventional multicast. From simulation results, we achieve that the proposed content distribution strategy offers significantly better performance.

1 Introduction

The development of computer and communication technologies has implemented the high-speed global network like Internet. And the improved coding algorithms have made a multimedia data smaller size. Especially multimedia system for on-Demand is the essential field of various applications; digital library information system, remote education, entertainments, and so on. On-demand system is characterized by the real-time operation and is needed the acceptable quality of streaming content from a server to clients. Most current Internet streaming applications require that a server maintains a large number of contents' streams and generates traffic whenever request is occurred. The limited network bandwidth has resulted in restriction of streaming services to clients, and it is serious problem in on-Demand system [1,2,3].

Proxy is a special kind of server located in between clients and server. The benefit of proxy caching mechanism is to reduce network traffic, average latency for fetching items, and the load of server. Since proxy can store items that have been transmitted, many requests can be directly served from the cache of proxy without generating

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1146 – 1155, 2006.

traffic from/to server. But proxies storing popular items are faced with heavier traffic than others [4,5,6,7]. As caches can influence whole network traffic, it is important to use the proper cache replacement algorithm which items are stored and which items are replaced. Today, most proxy systems use some kind of the Least-Recently-Used (LRU) replacement algorithm. The advantage of LRU is its simplicity; the disadvantage is that it might not achieve the best hit-ratio since it does not use item sizes, latency, and the frequency of request as popularity [13].

Multicast delivery technique is to minimize the required network bandwidth. During the predefined multicast grouping period T_m, service requests on the identical item are grouped into a multicast. It therefore can reduce the server's load and use network resource more efficiently. But there is still shortage of infrastructure for multicast on the Internet and hard to serve interactive functions [8,9,10]. It allows clients to share video streams being transmitted through one channel, where channel is the unit of network bandwidth needed to transmit one video stream. Transmission techniques for multicast fall in two categories: batching and patching [11,12]. In batching, the requests for the same video are delayed for a certain amount of time T_m to serve as many requests as possible with one multicast channel. It is regularly generated at every T_m when there are the requests for the same video within T_m. So, some of clients should wait until new multicast is generated.

This paper presents content distribution strategy which adopts the equivalent-loaded proxy caching mechanism to reduce the network bandwidth and server's load, and to prevent the identical video from duplicating among proxies. We have developed an end-to-end client/server architecture consisted of several Head-End-Nodes, and resided near to clients. Under the new scheme, streaming contents are always conveyed from content server through Switching Agent (SA) and/or HEN to clients, thus HENs are able to intercept and cache these streams. Content server delivers the requested video to clients through multicast not concerned with the number of requests. Server calculates the request frequency on each video for replacing caches in HENs and slices streams being transmitted into equal-sized pieces called segment by the predefined multicast interval and delivers sequentially these segments to HENs according to the request order. Therefore it prevents the segment of the same video from duplicating among HENs.

The remainder of the paper is organized as follows: In section 2 we describe the proposed multicast network using equivalent-loaded web caching mechanism. In section 3, we present the simulations and analysis of the results. Finally, we give out conclusion in section 4.

2 Multicast Network Model

The proposed multicast technique is operated under the web proxy caching mechanism for Head-End-Network (HNET) composed of several HENs adopted some caches. In this scheme, most of the communication bandwidth of the server is organized into a set of logical channels to transmit contents, we consider contents as video, and each of them is capable of transmitting a video at its own playback rate. The rest bandwidth of it is used for control of service requests and service notifications.

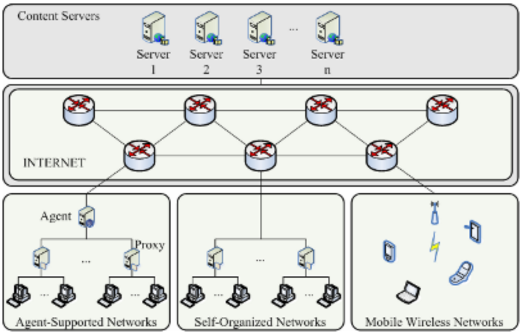


Fig. 1. The Structure diagram of web-cached multicast networks

Figure 1 shows the structure diagram of web-cached multicast networks that have three kinds of client side networks; agent-support networks, self-organized networks and mobile wireless networks. Agent-support networks are operated under the control of switching agent which manages all traffic from content server to HENs attached itself. Because there is not SA in self-organized networks, HENs should manage traffics among them. Mobile wireless networks are self-organized networks that should consider nodes' mobility. These networks are not easy to implement proxy caching strategy due to lack of processing power, limited energy, and variable network topology. This is out of our consideration.

The content server performs immediately multicast streaming service through Internet as a source device that transmits the first segment of each content only once except HENs take cache miss on it. The HNET operates under the control of server and shares its information among HNETs.

2.1 Content Distribution Strategy in Agent-Support Networks

Switching agent establishes a transmission channel from client to server when a particular video V_i is requested first and calculates the request frequency D_{pop} on it. If the requested video has already transmitted and stored among HENs, SA dose not request the transmission for an identical video except it is cache miss both in SA and in HENs. SA stores streams delivered from server depending on the D_{pop} and splits received streams into segments S_{number}, the value of the segment number is equal to Consecutive Value (CV), on an identical video within the predefined multicast grouping period T_m and transmits a segment to one of HENs according to the request order. Thus, different segments of streams on an identical video requested from several HENs are distributively stored at corresponding HENs.

To distribute segments orderly, SA collects the requests on an identical video with Video Identification (VID) within T_m, makes the transmission order depending on the request time. And then, SA inserts CV to each segments' header and transmits them orderly. This information indicates stored segments at each HEN classified by Node Identification (NID) and allows a HEN to find which HEN stores the rest parts of the requested video. To reduce processing time at each HEN, SA makes Multimedia Content Table (MCT) that has three elements; VID, CV and NID. For cache replacement, SA calculates how many segments will be transmitted to each HENs during next T_m and inserts this value (number of segments on each HENs: NS_{NID}) to MCT. It periodically transfers MCT to all HENs within its own HNET. Thus, all HENs under the control of SA can learn what they store and share their stored segments on an identical videos.

Stored segments in HEN may be deleted if a HEN's cache has been exhausted and this results in retransmitting deleted segments to the HEN from VOD server. To minimize the retransmission, HENs need much more cache but this is not cost-effective method. To reduce the capacity of HEN's cache, SA has some caches to store popular videos depending on the request frequency on an identical video content.

2.2 Content Distribution Strategy in Self-organized Networks

VOD server controls connection between client and HENs when i'th video V_i is requested first and calculates the request frequency F_i on it. Let all videos be ranked in order of their popularity where video i is the i'th most popular video. Server splits streams into segments $S_{i,\,j}$ by the predefined multicast interval T_m where j is j'th segment of video V_i and playback order O_j increases up to the number of HENs k by 1, and transmits a segment $S_{i,\,j}$ to one of HENs H_k according to the request order. Thus, different segments of streaming the identical video V_i requested by several HENs are stored at corresponding HENs. The size of the segment $S_{i,\,j}$ is equal to the predefined multicast interval T_m and HEN stores only one segment on video V_i. So the amount of stored segments on video V_i among HENs depends on the number of configured HENs.

To distribute segments orderly, server collects the requests on each video V_i within T_m and makes the playback order O_j as a First-In First-Out. And then, server inserts playback orders O_j to each segment's header and transmit streaming videos among HENs orderly. This information indicates what video segments are stored at each HEN and allows clients to find which HEN caches the rest parts of the requested video V_i. For cache replacement, server calculates how many segments will be transmitted to each HEN, N_{Hk} for H_k, during next multicast interval T_{m+1}. To reduce processing time at each HEN, server makes Switching Table (V_i, O_j, H_k) and transfers it to all of HENs within its own HNET at every T_m. Thus, all HENs under the control of server can learn what they store and share stored segments of the same video.

As an example, let us consider the following scenario in Figure 2. In $T_m \leq t < 2T_m$, clients in H_1 and H_5 request the video V_i expressed as playback_order(playback time, stored HEN) = (T_m, (H_1, H_5)) where the number of HENs k is 10. If it is the first time to request on video V_i, transmission order O_j is 1 and 5. The segment $S_{i,\,1}$ of video V_i will have been stored in H_1 at time $2T_m$, the segment $S_{i,\,2}$ in H_5 at time $3T_m$. within next T_m, request is ($2T_m$, (H_1, H_6, H_{10})), the updated playback order O_j is 1, 5, 6 and 10. The segment $S_{i,\,3}$ of video V_i will have been stored in H_6 at time $4T_m$, the segment $S_{i,\,4}$ in H_{10} at time $5T_m$. Request ($3T_m$, (H_3, H_5, H_6, H_{10})), the updated playback order O_j is 1, 5, 6, 10 and 3. The segment $S_{i,\,5}$ of video V_i will have been stored in H_3 at time $6T_m$, and so on. As a result, the first segment of video V_i for $0 \leq t < T_m$ is stored at H_1 (T_m, H_1), the second segment of video V_i for $T_m \leq t < 2T_m$ at H_5 ($2T_m$, H_5) and the rests are ($3T_m$, H_6), ($4T_m$, H_{10}), ($5T_m$, H_3), ($6T_m$, H_2), ($7T_m$, H_8),

($8T_m$, H_7), ($9T_m$, H_9) and ($10T_m$, H_4), respectively. After $10T_m$, the amount of stored segments among HENs is $10T_m$. To playback video V_i continuously, client must connect among HENs in the order of j. If clients who requests video V_i in $T_m \leq t < 2T_m$, series of playback_order are (T_m, H_1), ($2T_m$, H_5), ($3T_m$, H_6), ($4T_m$, H_{10}), ($5T_m$, H_3), ($6T_m$, H_2), ($7T_m$, H_8), ($8T_m$, H_7), ($9T_m$, H_9) and ($10T_m$, H_4), respectively. Because HEN can store only one T_m for each video V_i, there is no stored segments which will be played after $10T_m$.

Tm	HEN #									
	1	2	3	4	5	6	7	8	9	10
1	▨				▨					
2	▨					▨				▨
3			▨		▨	▨				
4		▨			▨			▨		▨
5					▨		▨			
6		▨					▨		▨	
7		▨				▨				
8	▨							▨		
9				▨						
10		▨		▨			▨			

(a) requests from HENs

Tm	playback order j									
	1	2	3	4	5	6	7	8	9	10
1	1	5								
2	1	5	6	10						
3	1	5	6	10	3					
4	1	5	6	10	3	2	8			
5	1	5	6	10	3	2	8	7		
6	1	5	6	10	3	2	8	7	9	
7	1	5	6	10	3	2	8	7	9	
8	1	5	6	10	3	2	8	7	9	
9	1	5	6	10	3	2	8	7	9	4
10	1	5	6	10	3	2	8	7	9	4

(b) Playback order

Fig. 2. Diagram for caching sequence depending on requests

To play the rest parts of the video V_i for V_i - $10T_m$, as there is no cached data among HENs, server generates a new multicast called dynamic multicast which has a variable starting time depending on the number of HENs that have cached the same video. If the amount of cached video is $n{\bullet}T_m$ where n is the number of HENs that have cached the same video, server should send the rest parts of video $|v|$ - $n{\bullet}T_m$ transmitted through dynamic multicast which is generated by server and which always starts at $n{\bullet}T_m$, where $|v|$ is the length of the video.. When dynamic multicast is generated, all clients should join dynamic multicast. So the interval of dynamic multicast is $\lfloor n \cdot T_m \rfloor$. Thus, client downloads the whole video $|v|$ with two different video streams through two channels. Since Client can download data on two channels simultaneously, one through unicast from one of HENs and the other through multicast from server.

In the above discussion, server sends video data in the following cases: 1) to cache among HENs by initial request and 2) to multicast the rest parts of the video as many as $|v|$ - $n{\bullet}T_m$. So, the total amount of video data being transmitted can be computed as D. D is equal to the summation of D_i and D_d, where D_i and D_d denote the mean total amount of data transmitted by the initial transmission and dynamic multicast, respectively. The total amount of video data delivered by the initial transmission, D_i, is equal to $n{\bullet}T_m$. If there is no change in transmission order O_t, as the data for playing the video v have been already cached among HENs, server does not need to send them. Since there are two kinds of dynamic multicast channels depending on the mean service requesting rate λ_v on video v denoted to requests/min, the amount of video data delivered by dynamic multicast D_d is

$$D_d = \begin{cases} \displaystyle\sum_{i=1}^{\left\lceil \frac{(|v|-n\cdot I_m)}{n\cdot I_m} \right\rceil} i \cdot n \cdot T_m, & where\ \lambda \geq \dfrac{1}{n} \\ \displaystyle\sum_{i=1}^{\left\lceil \frac{(|v|-n\cdot I_m)}{\tau} \right\rceil} i \cdot n \cdot T_m, & where\ \lambda < \dfrac{1}{n} \end{cases} \tag{1}$$

Since the mean interval of two consecutive dynamic multicast is n_v•T_m + τ_v for each video *v*, where $\tau_v = 1/\ \lambda_v$, the mean network bandwidth requirement of server B_s for *N* videos is

$$B_s = \sum_{v=1}^{N} \frac{D_{dv}}{|v_v| - n_v \cdot T_m} \cdot r_v \tag{2}$$

r_v is the playback rate of video *v*. D_{dv} are the amount of data for video *v* delivered by dynamic multicast.

3 Simulation and Analysis of the Results

In this section, we show simulation results to demonstrate the benefit of proposed multicast network with equivalent-loaded web caching mechanism and analyzes on the results of performance using it. We assume that the system contains 1000 videos, V_i =1000; all of them are 100 minutes long, |*v*| = 100T_m. The server is capable of supporting 10,000 channels and the total number of service request is limited 10,000 within 100 minutes. Let *N* be the total number of Videos in the Server. Let $P_N(i)$ be the conditional probability that, given the arrival of a video request, the arriving request is made for video i. Let all the videos be ranked in order of their popularity where video *i* is the *i*'th most popular video. We assume that $P_N(i)$, defined for *i* = 1, 2, …, N, has "cut-off" Zipf-like distribution given by

$$P(i) = \frac{\Omega}{i^Z}, \qquad \text{where } \Omega = \left(\sum_{i=0}^{N} \frac{1}{i^Z} \right)^{-1} \tag{3}$$

In this paper, we consider a broader class of distribution functions with exponents in *skew factor z* [7,13,14]. A larger z corresponds to a more severe skew condition indicating that some videos are requested more frequently than the others. We set this value to 0.7 [14]. Using the Zipf-like distribution, if the overall clients' service request rate to the VOD server is λ, the service requesting rate for *i*'th video is $\lambda i = \lambda P_N(i)$. Its rate based on popularity is used to determine the traffic for each video. The most popular video (*i*=1) must have higher weighted value than the others because the request for the most popular video is more frequent than that for unpopular ones. We use $P_N(i)$ as a weighting parameter for selection of videos in simulation and the service request rate λ follows Poisson distribution.

Consequently, we perform simulation such that the more popular videos with higher request probability [15]. Our workload and system parameters are summarized in Table. 1. The default values are listed under the *Default* column. We also vary some of these parameters to do sensitivity analysis. The ranges of values used for simulation are given in the third column under the *Range*.

Table 1. Parameters used for the simulations

Parameter	Default	Range
Number of videos	1000	100 ~ 2000
Video length (minutes)	100	N/A
Server bandwidth (streams)	10000	N/A
HEN bandwidth (streams)	100	N/A
Request rate λ (requests/min)	50	10 to 200
Skew factor z	0.7	N/A
Cache size in HEN (minutes)	100	100 to 1000
Number of HENs	10	1 ~ 20

Figure 3 shows the mean number of transmission channels from VOD server. In this case, the number of HENs was fixed at 10, server's bandwidth was 10,000, HEN bandwidth was 500 channels, and the mean request rate λ was 10 and 50 requests per minute. The cache size of HEN is $100I_m$. So the total cache size of HENs is $1000I_m$. All of transmission channels from VOD server are to transmit initial transmission and/or to transmit dynamic multicast after n• T_m of the video V_i. Those multicasts start playing after finishing the playback of all the segments cached among HENs. So, it can reduces the number of multicast to transmit video segments as many as n• T_m for each video.

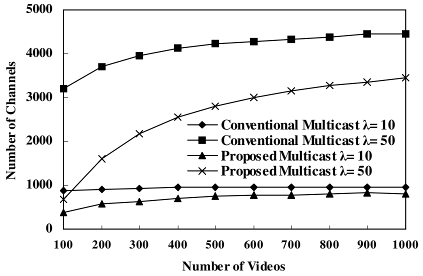


Fig. 3. The mean number of multicast channels for VOD server at arrival rate λ= 10 and 50

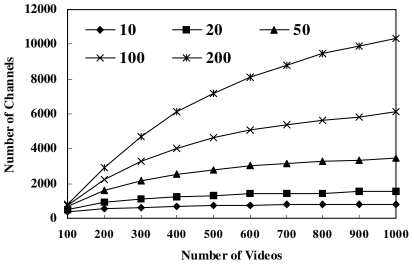


Fig. 4. The mean number of multicast channels in server as the function of the number of videos at various arrival rates

In proposed multicast technique, the multicast grouping interval T_m is various from T_m to n• T_m, where n is the number of HENs that cached the video V_i. We can reduce the number of multicast channels almost 59% for 100 videos and 15% for 1,000 videos at the mean arrival rate $\lambda = 10$ in compare with conventional multicast. At $\lambda = 50$, it can reduce 80 % for 100 videos and 22% for 1,000 videos. If the number of video is 100 and 1,000, cache can store as many as 1% and 0.1% of the total amount of video. So, as cache hit ratio of 1,000 videos is lower than 100 videos', server has to send more streams for clients to play uncached, called cache miss, video data. The number of multicast channels depends on cache hit ratio. Therefore, the number of transmission channels on server increases logarithmically because cache hit ratio grows logarithmically. Figure 4 shows simulation results of proposed multicast technique that the mean number of multicast channels in server as the function of the number of videos in various arrival rate λ 10, 20, 50, 100 and 200, and the others parameters were used the same ones in above simulations. This indicates that the number of channels in server grows depending on the number of videos logarithmically. It also grows depending on the arrival rate λ linearly.

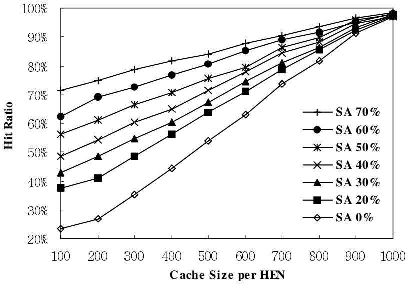


Fig. 5. Total cache hit ratio of HENs and SA for varying size of cache in HEN

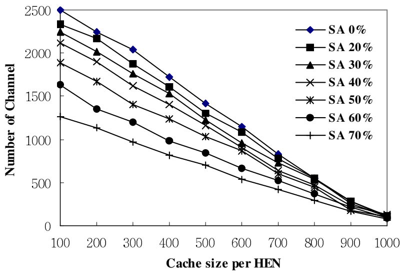


Fig. 6. The number of server's channels as the function of the size of cache

Figure 5 shows the total cache hit ratio of HENs and SA. In this case, the number of HENs was fixed at 10, the number of clients in each of HENs was 500, Server's bandwidth was 100, HNET bandwidth was 500 and the average request rate λ was 50 requests per minute. The cache size of HEN is in the range of 100 to 1,000 and SA stored popular videos requested 0% to 70% of the total request. We configure HNET one SA, 10 HEN and 500 clients in each of HENs. We observe that SA storing popular videos offers better cache hit ratio than not storing. In case that HEN has a little cache, the more SA stores popular videos the better cache hit ratio it acquires. This is due to the fact that popular videos are frequently requested more than the others. When SA stores popular videos (2 videos as long as 200 minutes) requested 20% of total requests and HENs store fragments of videos 1,000 minutes long, total hit ratio is almost 37%. But in case that no SA store videos and HEN store videos 3,000 minutes long, total hit ratio is below 35% though the capacity of HEN's cache grows 300%. In this case, the efficiency of cache is improved more than 250%. This shows that storing some popular videos in SA is better than increasing the capacity of HEN's cache. The number of transmission channels from VOD server under the same simulation environment as Figure 5 is shown in Figure 6. All of transmission channels from VOD server are for transmitting initial segments and/or for retransmitting cache missed segments. If HEN has sufficient cache space, there is little effect originated by stored popular videos at SA as HEN's cache can store all of transmitted videos.

Figure 7 and figure 8 show simulation results that the load of each HEN according to the number of videos and the number of HENs, respectively. In figure 7, the HEN's capability to cache each video is 1 minute long and the total amount of HEN's cache is 100 minutes long. So, the caching ratios depending on the number of videos 100, 500, 1000 and 2000 are 10%, 2%, 1% and 0.5%, respectively. When the number of videos is 100, the load of each HEN is almost same as 10%. But 1000 videos and 2000 videos, there are some variation due to the popularity of videos. The more the number of videos is, the more the number of unpopular videos is increased. This means the decrease of the probability of caching among HENs. Therefore HEN has cached unpopular videos has higher load than the others. This shows that the caching ratio is main factor to determine the load of HENs.

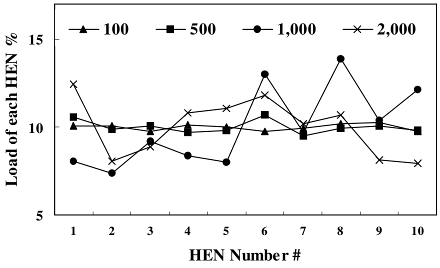


Fig. 7. Load of each HEN according to the number of videos varying 100 to 2000 at arrival rate $\lambda = 50$

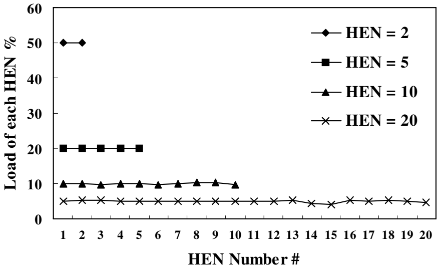


Fig. 8. Load of each HEN according to the number of HENs varying 2 to 20 at arrival rate $\lambda = 50$

In figure 8, the total amount of cached data is fixed at 1000 minutes long. So, the cache sizes of each HEN depending on the number of HENs 2, 5, 10 and 20 are 500, 200, 100 and 50 minutes long, respectively. Since the number of videos is 100 and the caching ratio is 10%. Simulation results show that the loads of HENs are well balanced and are almost (1/ the number of HENs). From the results, we acquire that proposed web-cached multicast technique has equivalent-loaded among HENs and it can reduce VOD server's bandwidth significantly.

4 Conclusion

Web-caching technique using proxy has been shown to be excellent techniques for reducing the demand on the server bandwidth. Unfortunately, it faces with an imbalanced traffic and load among proxies. Multicast delivery technique is one of the best ways to minimize the required bandwidth through the Internet. In this paper, we proposed the multicast technique using HNET which exists near clients and stores some segments of contents transmitted from server. HEN is able to cache the received segments and every HEN stores only one segment as much as multicast grouping interval for each content. The total amount of stored segments depends on the number of HENs. Client is served by stored segments among HENS according to playback order. The proposed technique makes the load of HENs evenly distribute over the configured network. We confirm the loads of distributive HENs composing a HNET are well balanced. With service request frequency, popular video are stored more frequently because they have higher probability of requests than others. So it can reduce the transmission channels of both server and HENs. From the simulation results, we can approximately reduce the number of multicast channels from server by 59%, 80% for 100 videos and by 15%, 22% for 1,000 videos at arrival rate λ is 10 and 50, respectively, compared with conventional multicast technique. So, the proposed multicast technique can improve server's workload, decrease the number of required multicast channels and distribute transmission channels over network even and orderly.

References

[1] T. Little, D. Vnekatesh, "Prospects for Interactive Video-on-Demand," IEEE Multimedia, pp.14-23, Fall, 1994.

[2] K. Almeroth, M. Ammar, "An Alternative Paradigm for Scalable On-Demand Applications: Evaluating and Deploying the Interactive Multimedia Jukebox," IEEE Transactions on Knowledge and data Engineering Special Issue on Web Technologies, April 1999.

[3] M.A. Goncalves, E.A.Fox, L.T.Watson, and N.A.Kipp, "Streams, Structures, Spaces, Scenarios, Societies (5S): A Formal Model for Digital Libraries," Virginia Tech Dept. of Computer Science Tech. Report TR-01-12, July 2001.

[4] A. Mahanti, C. Williamson, "Web Proxy Workload Characterization," Technical Report Univ. of Saskatchewan, Feb 1999.

[5] R. Rajaie, M. Handley, H. Yu, D. Estrin, "Proxy caching mechanism for multimedia playback streams in Internet," in 4th Int'l WWW Caching Workshop, Mar 1999.

[6] R. Rajaie, H. Yu, M. Handley, D. Estrin, "Multimedia Proxy Caching Mechanism for Quality Apdative Streaming Applications in the Internet," Proceeding of IEEE Infocom, Tel-Aviv, Israel, 2000

[7] P. Cao and S. Irani, "Cost-aware WWW Proxy Caching Algorithms," In Proc. Of the 1997 USENIX Symposium on Internet Technology and Systems, pp193-206, Dec 1997

[8] J. Pasquale, G. Polyzos, G. Xylomenos, "The Multimedia Multicasting Problem," ACM Multimedia Systems, vol. 6, No. 1, 1998

[9] Kien A. Hua, Ying Cai, Simon Sheu, "Patching: A Multicast Technique for True Video-on-Demand Services," ACM Mutimedia'98, pp.191-200, Bristol, UK, 1998.

[10] Wanjiun Liao and Victor.O.K. Li, "The Split and Merge Protocol for Interactive Video-on-Demand," IEEE Multimedia, pp.51-62, 1997

[11] C. Griwodz, M. Zink, M. Lieport, G. On, and R. Steinmetz, "Multicast for Savings in Cache-Based Video Distribution", Multimedia Computing and Networking, San Jose, CA, Jan 2000

[12] K. A. Hua, D. A. Tran, and R. Villafane, "Caching Multicast Protocol for On-Demand Video Delivery", Proc. of SPIE Multimedia Computing and Networking 2000, Vol. 3969, pp.2-13, Dec 1999

[13] Carey Williamson, "On Filter Effects in Web Caching Hierarchies," ACM Transactions on Internet Technology, Vol. 2, No. 1, pp. 47-77, February 2002.

[14] L. Breslau, P.Cao, L, Fan, G. Phillips, S. Shenker, "Web Caching and Zipf-like Distributions: Evidence and Implications," in Proceeding of the Conference on Computer Communications (IEEE Infocom), New York, Mar 1999.

[15] S.Gribble and E. Brewer, "System Design Issues for Internet Middleware Services: Deductions form a Large Client Trace," In Proceeding of the 1997 USENIX Symposium on Internet Technology and Systems, December, 1997.

Load Distribution Strategies in Cluster-Based Transcoding Servers for Mobile Clients[*]

Dongmahn Seo, Joahyoung Lee, Yoon Kim, Changyeol Choi,
Hwangkyu Choi, and Inbum Jung

Department of Computer Science and Engineering,
Kangwon National University, Chuncheon, 200-701, Korea
{sarum, jinnie4u, yooni, cychoi, hkchoi, ibjung}@kangwon.ac.kr

Abstract. The recent advance in wireless network technologies has enabled the streaming media service on the mobile devices such as PDAs and cellular phones. Since the wireless network has low bandwidth channels and mobile devices are actually composed of limited hardware specifications, the transcoding technology is needed to adapt streaming media to the given mobile devices. When large scale mobile clients demand the streaming service, load distribution strategies among transcoding servers highly impact on the total number of QoS streams. In this paper, the resource weighted load distribution strategy is proposed for the fair load balancing and the more scalable performance in cluster-based transcoding servers. Our proposed strategy is based on the weight of resources consumed for transcoding to classified client grades and the maximum number of QoS streams actually measured in transcoding servers. The proposed policy is implemented on cluster-based transcoding system. In experiments, we evaluate its fair load distribution and scalable performance according to the increase of transcoding servers.

1 Introduction

Based on recently the amazing growth of telecommunication, computer and image compression technologies, the streaming media service has been spotlighted in many multimedia applications. The large amount of network traffics and the high performance computing ability are inevitable to support the QoS streams [1, 2, 3]. However, since the wireless network has low bandwidth channels and many mobile devices compose of limited hardware specifications, the transcoding technology is needed to adapt the originally encoded MPEG media to the given mobile devices.

[*] This work was supported by Kangwon Institute of Telecommunications and Information, was supported by grant No. B1220-0401-0203 from the University Fundamental Research Program of the Ministry of Information & Communication in Republic of Korea and was supported by Basic Research Program of the Korea Science Engineering Foundation(KOSEF).

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1156–1165, 2006.

The transcoding system is usually composed of both the multimedia server with the originally encoded MPEG media and the transcoding servers to perform the adapting to the given environment. The multimedia server retrieves the MPEG media and sends them to the selected transcoding server. The transcoding server performs the transcoding to original MPEG video and also sustains the streaming service to the corresponding client. In particular, to provide QoS for clients, it is inevitable to guarantee streaming media without ceasing and jittering phenomena [3, 4, 5, 6, 7].

In this paper, the load distribution strategies for transcoding jobs are studied in cluster based servers. The cluster server architecture has an advantage of the ratio of performance to cost and is easily extended from the general PCs [5]. This model usually consists of a front-end node and multiple backend nodes. In our research, the front-end node is used as a load distribution server and the backend nodes work as transcoding servers. Based on load distribution strategies, the load distribution server distributes the transcoding requests of clients into transcoding servers. To provide the QoS streams for various kinds of mobile clients, we propose the Resource Weight Load Distribution (RWLD) strategy in the cluster-based transcoding servers. For the criteria of load distribution, we measure both the actual amounts of resources consumed and the maximum number of QoS streams by transcoding grades in each transcoding server. From the load weights by transcoding grades, the intrinsic property of streaming media can be reflected in the load distribution mechanism. And also, the two types of measured information are utilized as the threshold point of admission control to guarantee QoS for all clients. The proposed strategy is implemented on cluster-based transcoding system together with other load distribution strategies. From our experiments, the RWLD strategy shows the fair load distribution in the heterogeneous transcoding servers and it leads to better performance scalability according to the increase of transcoding servers.

The rest of this paper is organized as follows. Sect. 2 describes related work for our research. In sect. 3, the RWLD strategy is proposed to achieve the fair load distribution and more scalable performance in cluster-based transcoding servers. Sect. 4 explains our actual experimental environment. In sect. 5, the performance of the RWLD strategy is evaluated and compared to other load distribution strategies. Sect. 6 concludes the paper.

2 Related Work

2.1 MPEG Profile

Mobile devices have their own the computing power, memory, network capacity. To adapt their working environment, the streaming media should be transformed from the original contents. There are MPEG media specifications to support the streaming media to mobile devices [8, 9]. Table 1 shows the MPEG profile composed of video size, frame rate, bit rate based on the operating environment of the streaming media service. As shown the Table 1, the MPEG media can be classified by 4 grades and each grade designates its own working mobile device.

Table 1. Specification of MPEG Profile

Grade	Video size	Frame rate	Bit rate (kbps)	Mobile device
SQCIF	128 X 96	15	50	Cellular phone
QCIF	176 X 144	15	70	PDA
CIF	352 X 288	26	100	Laptop PC
4CIF	704 X 576	30	200	Desktop PC

2.2 Load Distribution Strategies

Many researches were undertaken for the load distribution strategies in cluster-based servers. In particular, the cluster-based server architecture has been utilized in the Web server, game server and file server areas. As representative strategies in these areas, there are RR(Round Robin), LC(Least Connection), WRR(Weighted Round Robin), DWRR(Dynamic Weighted Round Robin) and so on.

The RR strategy allocates servers according to the sequence of job arrival. Since the RR does not consider the state of servers and the intrinsic features of jobs, it is difficult to attain the effective load balancing among servers. The LC strategy uses the count of clients connected to each server. This strategy chooses the server with the least count value. The WRR strategy designates the different weight to each server based on the capability of servers. This approach can not reflect the state of servers dynamically changed. To address the problem, the DWRR strategy is suggested. For jobs distributing to servers, this strategy considers the current state of backend servers.

3 Resource Weight Load Distribution Strategy

To provide the QoS streams for various kinds of mobile clients, we propose the Resource Weight Load Distribution (RWLD) strategy. For the RWLD strategy, the actual amounts of resources consumed for transcoding should be measured on the individual transcoding servers by the grades of mobile device. After that, the maximum numbers of QoS streams by transcoding grades are measured on each transcoding server. Based on the two types of measured information, the RWLD strategy manages the fair load balancing among heterogeneous cluster servers as well as provides the scalable performance according to the increase of transcoding servers.

3.1 Resource Consumption by Transcoding Grades

To find the actual amount of resources consumed for each transcoding grade, we measure the usage of CPU, memory and network bandwidth exhausted by the classified grades described in the Table 1. A Desktop PC has a role for a transcoding server which is composed of 1.4 GHz CPU, 256 Mbytes Memory, and 100 Mbps Network Bandwidth. The Linux operating system is deployed and the FFMPEG program is used for the transcoding of MPEG media [4].

Table 2 shows the experimental results for transcoding 10 4CIF grade movies into SQCIF, QCIF and CIF grade respectively. As experimental results, we find that the transcoding for the same grade results in the almost same resource consumption rates regardless of which movies are selected. As shown in this Table, the CPU consumption rate is the highest among all resources. Based on the constant resource consumption rates, the resource weight for the corresponding transcoding grade can be computed in each transcoding server.

Table 2. Resource Consumption Rates by Transcoding Grades

Grage	CPU (%)	Memory (Mbytes)	Network (Kbps)
SQCIF	8.3	5.7	50
QCIF	8.5	5.8	70
CIF	16.3	6.4	100

3.2 Resource Weight Table

Under the RWLD strategy, the load distribution server uses the Resource Weight Table (RWT) for the fair load balancing and the admission control for guaranteed the QoS. The RWT is composed of 4 items such as the resource weight, maximum streams, total resource weight and accumulated weight. The first item means the relative resource consumption weights by transcoding grades. It is driven by the fastest exhausted resource when each transcoding server transcodes the original MPEG media into the corresponding grades. Table 3 shows the pseudo codes for computing the relative weight of transcoding grades in each transcoding server. M is the available memory in a transcoding server. B is the available network bandwidth. C is the available CPU capacity. Using the index i for the transcoding grade, Qci, Qri and Qmi are denoted as the CPU usage, network usage and memory usage for the corresponding transcoding grade i. For example, if we have 4 grades such as SQCIF, QCIF, CIF, 4CIF, the notations of CPU usages are Qc1, Qc2, Qc3 and Qc4 respectively. And also, the Wn means the relative resource consumption weight for transcoding grade n.

Since the firstly exhausted resource restricts the total number of transcoding requests, the RWLD strategy uses its property to compute the resource weight W_n. As shown in the following pseudo codes, the resource weight W_n for each transcoding grade is determined by the firstly exhausted resources. The C/Q_{ci}, M/Q_{mi}, B/Q_{ri} designate the number of transcoding requests under available the CPU capacity, the memory space and the network bandwidth respectively. Among them, the smallest number determines the relative resource weight W_n of all transcoding grades in the corresponding server. If the CPU is the firstly exhausted resource in a transcoding server, the equation (1) of the Table 3 is chosen to compute the relative resource weights. After that, the results are recorded into the first item to the corresponding server in the resource weight table, as shown as Table 4.

Table 3. Pseudo Codes for Resource Weight Computation

```
if ( M/$Q_{mi}$ $\geq$ B/$Q_{ri}$ $\geq$ C/$C_{mi}$ ) { // CPU is exhausted firstly
$W_n = \frac{Q_{C_n} \times 100}{\sum_{k=1}^{i} Q_{C_k}}$ (n=1,2,...,i) — equation (1)
}
else if ( B/$Q_{ri}$ $\geq$ C/$Q_{ri}$ $\geq$ M/$Q_{mi}$ ) { // Memory is exhausted firstly
$W_n = \frac{Q_{m_n} \times 100}{\sum_{k=1}^{i} Q_{m_k}}$ (n=1,2,...,i) — equation (2)
}
else if ( C/$Q_{ci}$ $\geq$ M/$Q_{mi}$ $\geq$ B/$C_{ri}$ ) { // Network is exhausted firstly
$W_n = \frac{Q_{r_n} \times 100}{\sum_{k=1}^{i} Q_{r_k}}$ (n=1,2,...,i) — equation (3)
}
```

Table 4. Snapshot of Resource Weight Table on Initial Stage

	Transcoding Server A			
	resource weight	maximum streams	total resoutce weight	accumulated weight
SQCIF	25	8	200	0
QCIF	35	7	245	0
CIF	40	6	240	0

The maximum streams means the maximum number of QoS streams by transcoding grades in each transcoding server. This value is also achieved throughout the actual measurement. The total resource weight is computed by multiplying the resource weight item and the maximum stream. This value represents the total resource weight guaranteed the QoS by transcoding grades in each transcoding server. The accumulated weight means the resource weight accumulated in the corresponding transcoding server by currently executing transcoding jobs. In initial stage, the accumulated weight is zero.

3.3 Load Balance and Admission Control

In the cluster-based server architecture, each server has the same hardware specifications or not. Using the heterogeneous transcoding servers, each server shows up different resource consumption rates during transcoding operations. In the RWT, the resource weight and accumulated weight items are exploited for the load balancing among heterogeneous transcoding servers. By looking at the performance of individual servers on the classified transcoding grades, the RWLD strategy can apply the fair load distribution to heterogeneous cluster-based transcoding servers.

To guarantee QoS to all serviced streams, the admission control is inevitable in the streaming media service. If a new transcoding request ruins the QoS for currently serviced all streams, the admission control should reject the new client request to protect the existing clients. In our RWLD strategy, the load distribution server performs the load balancing role as well as the admission control mission.

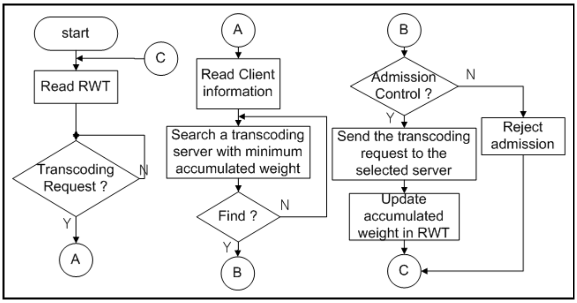

start
C
Read RWT
Transcoding Request ?
N
Y
A
A
Read Client information
Search a transcoding server with minimum accumulated weight
Find ?
N
Y
B
B
Admission Control ?
N
Y
Send the transcoding request to the selected server
Reject admission
Update accumulated weight in RWT
C

Fig. 1. Flow Chart of RWLD Strategy

Fig. 1 is the flow chart of the load balancing and the admission control in the RWLD strategy. As shown in this figure, the load distribution server initializes the RWT information and waits for client requests. To every transcoding requests, the RWLD strategy searches a transcoding server with the minimum accumulated weight so that the fair load balancing can be maintained. In addition, to guarantee the QoS for currently serviced streams, the RWLD strategy performs the admission control to the new transcoding request. If the admission is accepted, the new client request is sent to the selected transcoding server and its accumulated weight is updated. However, if the accumulated weight including the new request is over the total resource weight of the selected transcoding server, it is regarded as not eligible state for guaranteeing the QoS. In this case, since the new client request can destroy the QoS for currently serviced all clients, the admission control rejects the new client request.

4 Experimental Environment

In our experiment, the transcoding servers are composed of the 3 kinds of cluster systems. Total number of transcoding servers is 23 nodes. The cluster 1, 2 systems have 8 nodes respectively and the cluster 3 has 7 nodes. All nodes within a cluster system have the same hardware specification but the cluster systems have different hardware specifications.

We use the yardstick program to measure the performance of our cluster-based transcoding servers [10]. The yardstick program consists of the *virtual load generator* and the *virtual client daemon.*

The virtual load generator is located in the load distributed server. It generates client's transcoding requests based on the 3 parameters such as the distribution of transcoding grades, client's preferences to movies and client's arrival rate. Among the mobile devices, since the cellular phone takes a larger portion, we apply the Zipf distribution with the skew factor 0.271 to the transcoding from 4CIF grade to SQCIF grade [11]. The movies used in the Sect. 3.1 are used in our experiments. We regard that the popularity of each movie also follows a Zipf distribution with the skew factor 0.271. To the client's arrival rate, we use the Poisson distribution with λ=0.25[10, 12]. The virtual client daemon locates in test-bed PCs for clients. Based on the MPEG profile specification of Table 1, the virtual client daemon measures the time elapsed for receiving the stipulated frame rate and bit rates of the requested transcoded movies. If the elapsed time is below 1 second, the virtual client daemon remains in an idle state until 1 second period passes.

5 Performance Evaluation

From the implemented cluster-based transcoding system, the performance of the RR, DWRR and RWLD strategies are measured. As performance metrics, we designate 2 metrics. The first is the amount of CPU consumed according to the increase of clients because the CPU is the fastest exhausted resource in our previous experiment. As a second metric, the total number of QoS streams is selected to evaluate the scalable performance of tested strategies.

5.1 CPU Consumption Rates

Fig. 2 shows the amount of CPU usage of transcoding servers under RR, DWRR, RWLD strategies. We used 23 transcoding servers involved in 3 kinds of cluster system. On account of space in this figure, we chose 2 transcoding servers from each cluster system. The A node and B node is from the cluster system 1. The C node and D node belongs to the cluster system 2. The E node and F node is from the cluster system 3.

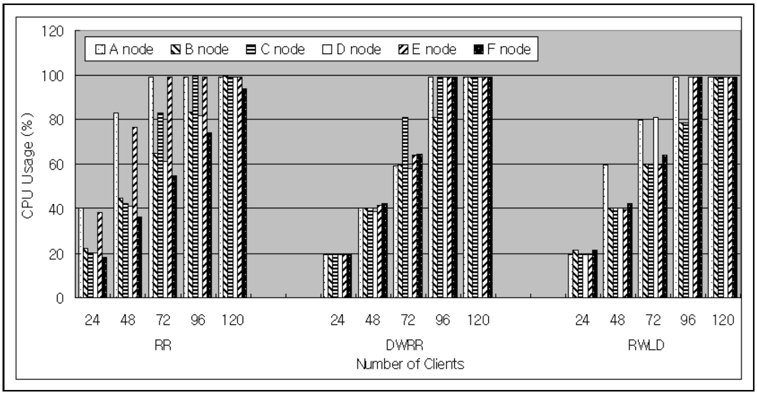


Fig. 2. RR (Round Robin) Strategy

As shown in the Fig. 2, the RR strategy results in the different amounts of CPU usage among transcoding servers. The reason is that transcoding jobs are distributed based on just the arrival order. In particular, since the RR strategy does not distinguish transcoding grades, it allows the overloaded transcoding servers and the underloaded servers to exist together. In the point of 120 clients, the CPU of the server A, C, E becomes saturate as 100% utilization rates, whereas the other servers do not.

In the DWRR strategy, transcoding servers send their current resource usages to the load distribution server by periodically. Based on this information, this strategy maintains the load balancing among transcoding servers. If the CPU usage of some transcoding servers reaches 100% utilization, this strategy does not require additional transcoding jobs to these servers. Since the workload congestion to some specific transcoding servers is avoided, the DWRR strategy does not destroy the QoS of all serviced streams. However, the load distribution server has overheads to communicate with transcoding servers. In addition, since the DWRR strategy uses just the CPU utilization rate as an admission control, it does not reflect the intrinsic characteristic of streaming media in real time requirement. Thus, even if the CPU utilization reaches 100%, the additional transcoding requests could be serviced to clients within the limited range. However, as shown in the Figure 5, the DWRR strategy shows fair load balancing among transcoding servers and does not ruin the QoS of all streams being serviced.

As shown in this Figure, the RWLD strategy maintains the fair load balancing among transcoding servers like the DWRR strategy. Since the DWRR uses the resource weights and the maximum streams according to the transcoding grades as the criteria of the load balancing and the admission control, there are no communication overheads between transcoding servers and the load distribution server. In addition, even if the CPU utilization reaches 100%, the additional transcoding jobs could be accepted in the range of proposed admission control mechanism. By considering the intrinsic property of streaming media, the RWLD strategy contributes the fair load balancing as well as the scalable performance in cluster-based transcoding servers.

5.2 Performance Scalability

Fig. 3 shows the total number of QoS streams supported by RR, DWRR, RWLD strategies accordingly as the number of transcoding servers is increased. The QoS is the most important mandatory requirement in the streaming media service. If the serviced streams are insufficient to guarantee the QoS requirement by transcoding grades, those streams can not involve in the total number of QoS streams. For our experiments, the load generator invokes 294 transcoding jobs. Under the Zipf distribution with 0.271 skew factor, the SQCIF grade is 44, the QCIF is 86 and the CIF is 64.

As illustrated in Fig. 3, the maximum number of clients increases proportional to the number of transcoding servers in all strategies. In the RR strategy, the overloaded servers with the congestion of transcoding jobs can not satisfy the QoS requirement. In particular, new transcoding requests allocated to the saturated servers has a negative impact on other QoS streams being serviced. From this reason, the RR strategy shows the relatively low performance improvement across the increase of transcoding servers.

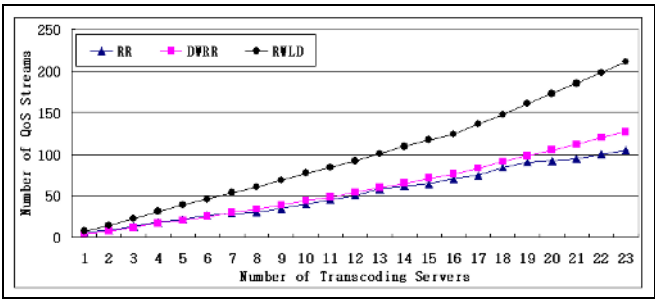


Fig. 3. Performance Scalability

The DWRR strategy does not consider the minimum amount of CPU consumed for transcoding to the desired transcoding grade. Even if the CPU utilization reaches 100%, it is possible to perform additional transcoding and streaming jobs within the range of satisfying the QoS requirement. The DWRR strategy does not consider this characteristic of streaming media. In addition, to monitor the CPU usages of transcoding servers, it has the communication overhead between transcoding servers and the load distribution server periodically. This overhead results in the further increase of the CPU usage in transcoding servers. As a result, the overhead itself and the failure to notice for the intrinsic property of streaming media have a negative impact on the performance scalability.

On the other hand, the RWLD strategy uses both the resource weight consumed and the maximum number of streams by transcoding grades as the criterion of the load balancing and the admission control. Based on these two types of pre-measured information, this strategy not only fully reflects the intrinsic property of streaming media but also has no communication overheads to monitor the state information of the resources in transcoding servers. Based on these advantages, even if the CPU utilization reaches 100%, the RWLD strategy can require the additional transcoding jobs within the range of satisfying the QoS requirement corresponding to each transcoding grade. As a result, the RWLD strategy has been the best scalable performance among the experimented load distribution strategies.

6 Conclusion

In this paper, the load distribution strategies are studied in the cluster-based transcoding servers. The load distribution strategy should provide the fair load balancing and scalable performance. We proposed the RWLD strategy used the actual amount of resources consumed by transcoding grades and the maximum number of QoS streams in transcoding servers.

In our heterogeneous cluster-based transcoding servers, we had evaluated the fair load balancing and the scalable performance of the RR, DWRR and RWLD strategies. The RWLD strategy maintained the fair load balancing among transcoding servers. This strategy used the resource weights and the maximum streams as the criteria of the load balancing and the admission control. By the two types of pre-measured information, this strategy not only reflects the intrinsic property of streaming media but also has no communication overheads to monitor the working state of transcoding servers. From our experiments, since the RWLD strategy performed the admission control based on the QoS requirements of the classified transcoding grades, it showed more linear performance scalability than other strategies.

References

1. Dinkar Sitaram, Asit Dan: Multimedia Servers: Applications, Environments, and Design. Morgan Kaufmann Publishers, 2000
2. W.C. Feng, M. Lie: Critical Bandwidth Allocation Techniques for Stored Video Delivery Across Best-Effort Networks. The 20th International Conference on Distributed Computing Systems, pp.201–207, 2000
3. D.H.C. Du, Y. J. Lee: Scalable Server and Storage Architectures for Video Streaming. IEEE Inter-national Conference on Multimedia Computing and Systems, pp.191–206, 1999
4. Florin Lahan, Irek Defee, Marius Vlad, Aurelian Pop, Prakash Sastry: Integrated system for multimedia delivery over broadband ip networks. IEEE Transactions on Consumer Electronics, Vol. 48, No.3, pp.564–565, 2002
5. C. Li, G. Peng, K. Gopalan, and T. Chiueh: Performance guarantees for cluster-based internet services, Proceedings of the 23rd International Conference on Distributed Computing Systems, pp378–385, May 2003
6. Sumit Roy, Michele Covell, John Ankcorn, and Susie Wee: A System Architecture for Managing Mobile Streaming Media Services, 23rd International Conference on Distributed Computing Systems Workshops (ICDCSW'03), pp.408–419, 2003
7. J. Guo, F. Chen, L. Bhuyan, and R. Kumar: A cluster-based active router architecture supporting video/audio stream transcoding services, Proceedings of the 17th International Parallel and Distributed Processing Symposium, pp.446–453, April 2003
8. http://www.mpeg.org
9. C. K. Hess, D. Raila, R.H. Cambell, and D. Mickunas: Design and performance of mpeg video streaming to palmtop computers, Proceedings of SPIE/ACM Multimedia Computing and Networking (MMCN2000), January 2000.
10. Brian K. Schmidt, Monica S. Lam, J. Duane Northcutt: The interactive performance of SLIM: a state-less, thin-client architecture. ACM SOSP'99, pp.31–47, 1999
11. C.C.Aggarwal, J.L.Wolf, and P.S.Yu: On optimal batching policies for viedo-on-demand storage servers, Proc. of IEEE ICMCS'96, pp.253–258, Hiroshima, Japan, June 1996
12. Surendar Chandra, Carla Schlatter Ellis and Amin Vahdat: Differentiated Multimedia Web Services Using Quality Aware Transcoding, Proceedings of IEEE INFOCOM Conference, March 2000

Safety of Recovery Protocol Preserving MW Session Guarantee in Mobile Systems*

Jerzy Brzeziński and Anna Kobusińska

Institute of Computing Science,
Poznań University of Technology, Poland
{Jerzy.Brzezinski, Anna.Kobusinska}@cs.put.poznan.pl

Abstract. In this paper checkpointing and rollback-recovery protocol rVsMW for mobile systems is presented. The protocol preserves Monotonic Writes session guarantee required by clients, despite failures of servers. The costs of rollback-recovery are minimized, by exploiting semantics of operations and properties of MW guarantee. The proof of safety property of rVsMW is included.

Keywords: rollback-recovery, safety, mobile systems, Monotonic Writes session guarantee.

1 Introduction

In the mobile environment, the replication of shared data is the key to obtaining high data availability, good access performance, and good scalability. Replication introduces, however, the problem of data consistency that arises when replicated objects are modified. This problem is directly related to the question of how results of operations, executed concurrently on different replicas of the same object (data, service), can be perceived.

The properties of distributed system concerning consistency depend in general on application and are formally specified by consistency models. The existing consistency models are not suitable for mobile systems, where clients accessing the data are not bound to particular servers and can switch from one server to another. Therefore, a new class of consistency models, called session guarantees, recommended for mobile environment, has been introduced [TDP+94]. Session guarantees, also called client-centric consistency models, define required properties of the system regarding consistency from the client's point of view. Four session guarantees have been defined: *Read Your Writes* (RYW), *Monotonic Writes* (MW), *Monotonic Reads* (MR) and *Writes Follow Reads* (WFR) and protocols implementing them have been introduced [BS05, BSW05b, BSW05a].

Because of dependability requirements of mobile applications, such consistency protocols should provide required session guarantees, despite severs' failures. But, as far as we know, none of proposed consistency protocols for mobile environment,

* This work was supported in part by the State Committee for Scientific Research (KBN), Poland, under grant KBN 3 T11C 073 28.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3983, pp. 1166–1174, 2006.

which preserves session guarantees, is fault-tolerant. The lack of consistency protocols optimized in terms of rollback-recovery, makes the construction of effective solutions adjusted to real applications requirements more difficult.

For this reason, in this paper a checkpointing and rollback-recovery protocol rVsMW, which preserves Monotonic Writes session guarantee is presented. The proposed protocol integrates the popular fault–tolerant techniques: logging and checkpointing with coherence operations of VsSG protocol. As a result, the rVsSG protocol offers the ability to overcome the servers' failures, at the same time preserving MW session guarantee. Because of client orientation, in rVsMW protocol run-time faults are corrected with any intervention from the user. The main contribution of this paper is a formal proof of safety of the rVsMW protocol.

2 System Model

Throughout this paper, a replicated distributed storage system is considered. The system consists of a number of unreliable *servers* holding a full copy of *shared objects* and *clients* running applications that access these objects. Clients are mobile, i.e. they can switch from one server to another during application execution. To access shared object, clients select a single server and send a direct request to this server. Operations are issued by clients synchronously, i.e. a new operation may be issued after the results of the previous one have been obtained.

Since clients are separated from one another, a crash of one client does not influence the processing of other clients. For that reason, in this paper we consider only failures of servers. We assume the *crash-recovery* model of failures, i.e. servers may crash and recover after crashing a finite number of times [GR04]. Servers can fail at arbitrary moments and we require any such failure to be eventually detected, for example by failure detectors [SDS99].

The storage replicated by servers does not imply any particular data model or organization. Operations performed on shared objects are divided into *reads* and *writes*. Reads do not change the state of objects, while writes may create a new object, delete the existing one or cause the update of the object state. The server, which first obtains the write from a client, is responsible for assigning it a globally unique identifier. Clients can concurrently submit conflicting writes at different servers, e.g. writes that modify the overlapping parts of data storage.

3 Notation and Basic definitions

Operations on shared objects issued by client C_i are ordered by a relation $\xrightarrow{C_i}$ called *client issue order*. Server S_j performs operations in an order represented by relation $\overset{S_j}{\rightarrowtail}$. Operations on objects are denoted by w, r or o, depending on the operation type (write, read or these whose type is irrelevant).

In the paper, it is assumed that clients perceive the data from the replicated storage according to Monotonic Writes session guarantee. MW session guarantee orders writes issued by a single client. A server, before accepting a new write from a client, must perform all previous writes requested by this client. Formally, MW session guarantee is defined as follows [Sob05]:

Definition 1. *Monotonic Writes (MW) session guarantee is a property meaning that:*

$$\forall C_i \, \forall S_j \left[w_1 \overset{C_i}{\rightharpoonup} w_2 |_{S_j} \implies w_1 \overset{S_j}{\rightarrowtail} w_2 \right]$$

In the paper, it is assumed, that data consistency is managed by the VsSG *consistency protocol* [BS05].

The underlying consistency protocol uses a concept of server-based version vectors for efficient representation of sets of writes required by clients and necessary to check on the server side. Server-based version vectors have the following form: $V_{s_j} = \left[v_1 \; v_2 \; ... \; v_{N_S} \right]$, where N_S is a total number of servers in the system and single position v_i is the number of writes performed by server S_j.

Every write in the VsSG protocol is labeled with a *vector timestamp*, set to the current value of the vector clock V_{S_j} of server S_j, performing the write for the first time. The vector timestamp of write w is returned by function $T : \mathcal{O} \mapsto V$. All writes performed by the server in the past are kept in set $\mathcal{O}_{S_j}$. On the client's side, vector W_{C_i}representing writes issued by client C_i is maintained.

The sequence of past writes is called *history.* A formal definition of history is given below:

Definition 2. *A history H_{S_j} at time moment t, is a linearly ordered set $\left(\mathcal{O}_{S_j}, \overset{S_j}{\rightarrowtail} \right)$ where $\mathcal{O}_{S_j}$ is a set of writes performed by server S_j, till the time t and relation $\overset{S_j}{\rightarrowtail}$ represents an execution order of writes.*

The VsSG protocol eventually propagates all writes to all servers. During synchronization of servers, their histories are *concatenated.* The concatenation of histories H_{S_j} and H_{S_k}, denoted by $H_{S_j} \oplus H_{S_k}$, consists in adding new operations from H_{S_k} at the end of H_{S_j}, preserving at the same time the appropriate relations [BS05].

Below, we propose formal definitions of fault-tolerance mechanisms used by the rVsMW protocol:

Definition 3. *Log Log_{S_j} is a set of triples:*

$$\left\{ \langle i_1, o_1, T(o_1) \rangle \; \langle i_2, o_2, T(o_2) \rangle \; ... \; \langle i_n, o_n, T(o_n) \rangle \right\},$$

where i_n represents the identifier of the client issuing a write operation $o_n \in \mathcal{O}_{S_j}$ and $T(o_n)$ is timestamp of o_n.

Definition 4. *Checkpoint $Ckpt_{S_j}$ is a couple $\left\langle V_{S_j}, H_{S_j} \right\rangle$, of version vector V_{S_j} and history H_{S_j} maintained by server S_j at the time t, where t is a moment of taking a checkpoint.*

Every server stores objects in the violate memory, whose content is lost when the failure occurs. However, for the sake of recovery procedure, it is commonly assumed that servers have also access to a stable storage, able to survive all failures [EEL+02]. The log and the checkpoint are saved by the server in the stable storage. Additionally, the newly taken checkpoint replaces the previous one, so just one checkpoint for each server is kept in the stable storage.

```
Upon sending a request $\langle o \rangle$ to server $S_j$ at client $C_i$

if iswrite($o$) then
    $W \leftarrow \max(W, W_{C_i})$
end if
send $\langle o, i, W \rangle$ to $S_j$

Upon receiving a request $\langle o, i, W \rangle$ from client $C_i$ at server $S_j$

while $\left(V_{S_j} \not\geq W\right)$ do
    wait()
end while
if iswrite($o$) then
    if $i \in CW_{S_j}$ then
        $secondWrite \leftarrow TRUE$
    else
        $CW_{S_j} \leftarrow CW_{S_j} \cup i$
    end if
    $V_{S_j}[j] \leftarrow V_{S_j}[j] + 1$
    timestamp $o$ with $V_{S_j}$
    $Log_{S_j} \leftarrow Log_{S_j} \cup \langle i, o, T(o) \rangle$
    perform $o$ and store results in $res$
    $H_{S_j} \leftarrow H_{S_j} \oplus \{o\}$
    if $secondWrite$ then
        $Ckpt_{S_j} \leftarrow \langle V_{S_j}, H_{S_j} \rangle$
        $Log_{S_j} \leftarrow \emptyset$
        $CW_{S_j} \leftarrow \emptyset$
        $secondWrite \leftarrow FALSE$
    end if
end if
if not iswrite($o$) then
    perform $o$ and store results in $res$
end if
send $\left\langle o, res, V_{S_j} \right\rangle$ to $C_i$

Upon receiving a reply $\langle o, res, W \rangle$ from server $S_j$ at client $C_i$

if iswrite($o$) then
    $W_{C_i} \leftarrow \max(W_{C_i}, W)$
end if
deliver $\langle res \rangle$

Every $\Delta t$ at server $S_j$
foreach $S_k \neq S_j$ do
    send $\left\langle S_j, H_{S_j} \right\rangle$ to $S_k$
end for

Upon receiving an update $\langle S_k, H \rangle$ at server $S_j$
foreach $w_i \in H$ do
    if $V_{S_j} \not\geq T(w_i)$ then
        perform $w_i$
        $V_{S_j} \leftarrow \max\left(V_{S_j}, T(w_i)\right)$
        $H_{S_j} \leftarrow H_{S_j} \oplus \{w_i\}$
    end if
end for
signal()

On rollback-recovery
$\langle V_{S_j}, H_{H_j} \rangle \leftarrow Ckpt_{S_j}$
$Log^{'}_{S_j} \leftarrow Log_{S_j}$
$vrecover \leftarrow \mathbf{0}$
foreach $o^{'}_j \in Log^{'}_{S_j}$ do
    choose $\langle i^{'}, o^{'}_i, T(o^{'}_i) \rangle$ with minimal
    $T(o^{'}_j)$ from $Log^{'}_{S_j}$ where $T(o^{'}_j) > V_{S_j}$

    $V_{S_j}[j] \leftarrow V_{S_j}[j] + 1$
    perform $o^{'}_j$
    $H_{S_j} \leftarrow H_{S_j} \oplus \left\{o^{'}_j\right\}$
    $CW_{S_j} \leftarrow CW_{S_j} \cup i^{'}$
    $vrecover \leftarrow T(o^{'}_i)$
end for
$secondWrite \leftarrow FALSE$
```

Fig. 1. Checkpointing and rollback-recovery rVsMW protocol

4 The rVsMW Protocol

The VsSG coherency protocol assumes that servers are reliable, i.e. they do not crash. Such assumption might be consider not plausible and too strong for certain mobile distributed systems. Therefore, the rVsMW protocol, which equips the VsSG protocol with fault-tolerance mechanisms, is proposed.

For every client C_i executing write w, results of all writes preceding w in a client issue order cannot be lost if MW is to be preserved. On the other hand, until write w is not followed by another write issued by the same client, then results of w are not essential for preserving MW. Unfortunately, at the moment of performing the operation, the server does not possess the knowledge, whether in the future a client will issue another write request, or not.

So, to preserve MW, the rVsMW protocol must ensure that results of all writes issued by the client are not lost in the case of server failure and its recovery. It is performed by logging in the stable storage the operation issued by a client and its timestamp. Additionally, the server state is checkpointed occasionally to bound the length of a message log. Logging and checkpointing operations are integrated with operations of VsSG consistency protocol.

The server, which obtains the write request directly from client C_i, logs the request to stable storage (Fig. 1, l. 17), if it fulfills MW (l. 6)[Sob05]. It is important that logging of write takes place before performing this request (l. 18). Such an order is crucial because, if the operation is performed but not logged, it could be lost in the case of subsequent failure.

The moment of taking a checkpoint is determined by obtaining a second write request from the same client (l. 21). Saving the state of server earlier is excessive, because the loss of write request that is not followed by another write, does not violate MW. Essential is the fact, that firstly the checkpoint is taken, and only afterwards the content of log Log_{S_j} is cleared. (l. 22).

After the failure occurrence, the failed server restarts from the latest checkpoint (l. 46) and replays operations from the log (l. 49-57) according to their timestamps, from the earliest to the latest one.

Writes received from other servers during update procedure, and missing from the local history of S_j, are not logged (l. 40-42). Thus, they are lost after the failure occurrence. However, by the assumption, such writes are saved in the stable storage (in the log or in the checkpoint) of servers, which received them directly from clients. Hence, lost writes will be obtained again in consecutive synchronizations.

5 Safety of rVsMW Protocol

Lemma 1. *Every write operation w issued by client C_i and performed by server S_j that received w directly from client C_i, is kept in checkpoint $Ckpt_{S_j}$ or in log Log_{S_j}.*

Proof. Let us consider write operation w issued by client C_i and obtained by server S_j.

1. From the algorithm, server S_j before performing the request w, saves it in the stable storage by adding it to log Log_{S_j} (l. 17). Because logging of w takes place before performing it (l. 18), then even in the case of failure operation w is not lost, but remains in the log.
2. Log Log_{S_j} is cleared after performing by S_j another write operation issued by the same client. However, according to the algorithm, the second write is logged (l. 17), before being performed (l. 18), and only afterwards the server's version vector V_{S_j} and history H_{S_j} are stored in the checkpoint $Ckpt_{S_j}$ (l.21). The operation of clearing log Log_{S_j} (l. 22) is made after the checkpoint is taken. Therefore, the server failure, which occurs after clearing the log, does not affect safety of the algorithm because writes from the log are already stored in the checkpoint.
3. After the checkpoint is taken, but before the log is cleared (between lines 21 -22) writes issued by client C_i and performed by server S_j are stored in both the checkpoint $Ckpt_{S_j}$ and the log Log_{S_j}.

Lemma 2. *The rollback-recovery procedure recovers all write operations issued by clients and performed by server S_j that were logged in log Log_{S_j} in the moment of server S_j failure.*

Proof. Let us assume that server S_j fails. The rollback-recovery procedure, after recovering V_{S_j} and H_{S_j} from a checkpoint (l. 46), recovers operations remembered in the log (l. 49). The recovered operation updates version vector V_{S_j} (l. 50), it is performed by S_j (l. 51) and added to the S_j's history H_{S_j}(l. 53).

Assume now, that failures occur during the rollback-recovery procedure. Due to such failures the results of operations that have already been recovered are lost again. However, since log Log_{S_j} is cleared only after the checkpoint is taken (l. 22) and it is not modified during the rollback-recovery procedure (l. 47), the log's content is not changed. Hence, the recovery procedure can be started from the beginning without loss of any operation issued by clients and performed by server S_j after the moment of taking checkpoint.

Lemma 3. *Operations obtained and performed in the result of synchronization procedure, are performed again after the failure of S_j before processing a new write from a client, if they are required by MW.*

Proof. By contradiction, let us assume that server S_j has performed a new operation w_2 obtained from client C_i before performing again operation w_1, received during a former synchronization and lost because of S_j failure. According to VsSG protocol, while executing w_2 the condition $V_{S_j} \geq W_{C_i}$is fulfilled (l. 6) .

Further assume, that w_1 issued by C_i before w_2, has been performed by server S_k. According to the algorithm, after the reply from S_k is obtained by C_i, vector W_{C_i} is modified: $W_{C_i} \leftarrow \max(W, W_{C_i})$. This means that vector W_{C_i} is updated at least at position k: $W_{C_i}[k] \leftarrow k+1$. (l. 32).

Server S_j, during synchronization procedure with S_k, performs w_1 and updates its version vector: $V_{S_j} \leftarrow \max\left(V_{S_j}, T(w_1)\right)$, which means that V_{S_j} has been modified at least in the position k (l. 41). However, if the failure of S_j

happens, the state of S_j is recovered accordingly to values stored in the checkpoint $Ckpt_{S_j}$ (l. 46) and in the log Log_{S_j} (l. 48-55). From the algorithm, while recovering operations from the log, the vector V_{S_j} is updated only at position j.

Thus, if operation w_1 performed by S_j in the result of synchronization with server S_k is lost because of S_j failure, the value of $V_{S_j}[k]$ does not reflects the information on w_1. Hence, until the next update message is obtained, $V_{S_j}[k] < W_{C_i}[k]$, which contradicts the assumption.

Lemma 4. *The server performs new write operation issued by a client only after all writes performed before the failure are recovered.*

Proof. By contradiction, let us assume that there is a write operation w performed by server S_j before the failure occurred, that has not been recovered yet, and that the server has performed a new write operation issued by client C_i. According to underlying VsSG protocol, for server S_j that performs the new write operation, the condition $V_{S_j} \geq W_{C_i}$is fulfilled (l. 6-8).

Let us consider which actions are taken when a write operation is issued by client C_i and performed by server S_j.

On the server's side, the receipt of the write operation causes the update of vector V_{S_j} in the following way: $V_{S_j}[j] \leftarrow V_{S_j}[j] + 1$ and results in timestamping w with the unique identifier (l. 16). The server, which has performed write, sends a reply that contains the modified vector V_{S_j} to the client.

On the client's side, after the reply is received, vector W_{C_i} is modified: $W_{C_i} \leftarrow \max(W, W_{C_i})$. This means that vector W_{C_i} is updated at least at position j: $W_{C_i}[j] \leftarrow j + 1$ (l. 32).

If there is write operation w performed by server S_j before the failure that has not been recovered yet, then $V_{S_j}[j] < W_{C_i}[j]$, which follows from the ordering of recovered operations (l. 50). This is a contradiction with $V_{S_j} \geq W_{C_i}$. Hence, the write operation cannot be performed until all previous writes are recovered.

Theorem 1. *MW session guarantee is preserved by rVsMW protocol for clients requesting it, even in the presence of server failures.*

Proof. Let us consider operations w_1 and w_2, issued by client C_i, which requires MW session guarantee. Let write operation w_2 follow write w_1 in the client's issue order and let w_2 be performed by server S_j.

It has been proved that the VsSG protocol preserves the MW session guarantee, when none of servers fails, i.e. for any client C_i requiring MW and for any server S_j the relation $\forall C_i \, \forall S_j \left[w\ _1 \xrightarrow{C_i} w_2|_{S_j} \Rightarrow w_1 \overset{S_j}{\rightarrowtail} w_2 \right]$ holds [BSW05b]. According to Lemma 1, every write operation performed by server S_j is saved in the checkpoint or in the log. After the server failure, all operations from the checkpoint are recovered. Further, all operations performed before the failure occurred, but after the checkpoint was taken, are also recovered (according to Lemma 2). All recovered write operations are applied before new writes issued by the client are performed (according to 4). Moreover, operations obtained by S_j during synchronization procedure, which are required by MW, and possibly were lost because of S_j failure, are also performed once again before new writes from C_i. Hence, for any client C_i and any server S_j, MW session guarantee is preserved by the rollback–recovery and checkpointing rVsMW protocol.

Full versions of theorems and proofs can be found in [BKK05].

6 Conclusions

This paper addresses a problem of integrating the consistency management of the mobile system with the recovery mechanism. We introduce the rollback-recovery protocol rVsMW which preserves Monotonic Writes session guarantee. A correctness proof, showing that the protocol is safe, i.e. MW is provided despite servers failures, is included.

The rVsMW protocol has features similar in general to pessimistic message logging. However, in contrast to message–passing systems, we consider the interaction between the client and the server, not between the servers. This is a novel feature, which follows directly from the session guarantees assumptions which are client–oriented. Moreover, in contrast to systems with message–passing, we also take into account the semantics of operations. This results in checkpointing only results of write operations which are essential to provide MW.

Our future work encompasses the development of rollback-recovery protocols, which preserve other session guarantees. Moreover, appropriate simulation experiments testing rVsMW protocol are being prepared.

References

[BKK05] J. Brzeziński, A. Kobusińska, and J. Kobusiński. Safety of rvsmw rollbackrecovery protocol for mobile systems. Technical Report RA-011/05, Institute of Computing Science, Poznań University of Technology, November 2005.

[BS05] J. Brzeziński and C. Sobaniec. Safety of an object-based version vector consistency protocol of session guarantees. In *Proc. of the 6th Int. Conf. on Parallel Processing and Applied Mathematics*, Poznań, Poland, September 2005.

[BSW05a] J. Brzeziński, C. Sobaniec, and D. Wawrzyniak. Safety of a client-based version vector consistency protocol of session guarantees. In *Proc. of the 19th Int. Symp. on Distributed Computing (DISC 2005)*, Cracow, Poland, September 2005.

[BSW05b] J. Brzeziński, C. Sobaniec, and D. Wawrzyniak. Safety of a server-based version vector protocol implementing session guarantees. In *Proc. of Int. Conf. on Computational Science (ICCS2005), LNCS 3516*, pages 423–430, Atlanta, USA, May 2005.

[EEL+02] N. Elmootazbellah, Elnozahy, A. Lorenzo, Yi-Min Wang, and D.B. Johnson. A survey of rollback-recovery protocols in message-passing systems. *ACM Computing Surveys*, 34(3):375–408, September 2002.

[GR04] Rachid Guerraoui and Luis Rodrigues. *Introduction to distributed algorithms.* Springer-Verlag, 2004.

[SDS99] N. Sergent, X. Dt'efago, and A. Schiper. Failure detectors: Implementation issues and impact on consensus performance. Technical Report SSC/1999/019, t'Ecole Polytechnique Ft'edt'erale de Lausanne, Switzerland, May 1999.

[Sob05] C. Sobaniec. *Consistency Protocols of Session Guarantees in Distributed Mobile Systems.* PhD thesis, Institute of Computing Science, Poznan University of Technology, September 2005.

[TDP+94] Douglas B. Terry, Alan J. Demers, Karin Petersen, Mike Spreitzer, Marvin Theimer, and Brent W. Welch. Session guarantees for weakly consistent replicated data. In *Proc. of the Third Int. Conf. on Parallel and Distributed Information Systems (PDIS 94)*, pages 140–149, Austin, USA, September 1994. IEEE Computer Society.

Author Index