

Gilbert Babin
Peter Kropf
Michael Weiss (Eds.)

LNBIP 26

E-Technologies: Innovation in an Open World

4th International Conference, MCETECH 2009
Ottawa, Canada, May 2009
Proceedings

 Springer

Lecture Notes in Business Information Processing

26

Series Editors

Wil van der Aalst

Eindhoven Technical University, The Netherlands

John Mylopoulos

University of Trento, Italy

Norman M. Sadeh

Carnegie Mellon University, Pittsburgh, PA, USA

Michael J. Shaw

University of Illinois, Urbana-Champaign, IL, USA

Clemens Szyperski

Microsoft Research, Redmond, WA, USA

Gilbert Babin Peter Kropf
Michael Weiss (Eds.)

E-Technologies: Innovation in an Open World

4th International Conference, MCETECH 2009
Ottawa, Canada, May 4-6, 2009
Proceedings

Volume Editors

Gilbert Babin
HEC Montréal
Technologies de l'information
Montréal (Québec), H3T 2A7, Canada
E-mail: Gilbert.Babin@hec.ca

Peter Kropf
Université de Neuchâtel
Institut d'informatique
2000 Neuchâtel, Switzerland
E-mail: Peter.Kropf@unine.ch

Michael Weiss
Carleton University
Department of Systems and Computer Engineering
Ottawa, Ontario, K1S 5B6, Canada
E-mail: weiss@sce.carleton.ca

Library of Congress Control Number: Applied for

ACM Computing Classification (1998): H.3.5, H.4, J.1, K.4.4

ISSN 1865-1348
ISBN-10 3-642-01186-1 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-01186-3 Springer Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

springer.com

© Springer-Verlag Berlin Heidelberg 2009
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12659616 06/3180 5 4 3 2 1 0

Preface

The Internet pervades many of the activities of modern societies and has become the preferred medium for the delivery of information and services. The successful implementation of Internet applications, ranging from eBusiness, to eEducation or to eGovernment, is a multi-faceted problem, involving technological, managerial, economic, and legal issues. The MCETECH Conference on e-Technologies has become a well-recognized forum bringing together researchers, decision makers, and practitioners interested in exploring the many facets of Internet applications and technologies, with a focus on the technological, managerial, and organizational aspects. MCETECH 2009 focused on the theme “Innovation in an Open World”, covering topics such as inter-organizational processes, service-oriented architectures, security and trust, middleware infrastructures, open source and open environments and applications including eGovernment, eEducation, and eHealth. The fourth edition of the International MCETECH Conference on e-Technologies was held in Ottawa, Canada, May 4–6 2009.

There were 42 contributions submitted to MCETECH 2009, 36 for the research track and 6 for the industrial track. All papers underwent a thorough refereeing process, where each submission was sent to at least three Program Committee members for review. The papers were judged based on relevance, originality, soundness, and presentation. After in-depth discussions, 27 high-quality contributions were selected for presentation at the conference and for publication in this volume. Nineteen long and four short research papers were selected as well as four papers for the industrial track presenting experience reports and lessons learned from the trenches.

As in previous years, the main scientific conference program of MCETECH 2009 was accompanied by a rich tutorial program and two workshops, respectively, on the practice and theory of IT security (PTITS 2009) and on system interoperability in healthcare systems (E-health: Towards System Interoperability through Process Integration and Performance Management).

We thank all the authors who submitted papers, the Program Committee members, and the external reviewers. We express our gratitude to the Steering Committee Chair, Hafedh Mili, for his enthusiasm and his invaluable help in preparing this conference.

We also thank all the local people who were instrumental in making this edition of MCETECH another very successful event. In particular, we are very grateful to Daniel Amyot, who was responsible for the local arrangements. Furthermore, we thank Liam Peyton and Morad Benyoucef, who organized industrial liaison, publicity, and sponsorship activities, and the many students who

volunteered on the organization team, as well as the IT services of the University of Ottawa and the Université du Québec à Montréal.

May 2009

Gilbert Babin
Peter Kropf
Michael Weiss

Organization

The 4th International MCETECH Conference on e-Technologies (MCETECH 2009) was sponsored by Talent First Network, the Telfer School of Business (University of Ottawa), Carleton University, Université du Québec à Montréal, HEC Montréal, and Université de Neuchâtel.

Conference Chair

Michael Weiss Carleton University, Canada

Steering Committee

Michelle Blanc Analyweb Inc., Canada
Ross Button VP Emerging Technologies, CGI, Canada
Teodor Crainic ESG - UQAM, Canada
Philippe Deschênes Mediagrif Inc., Canada
Mark Fox University of Toronto, Canada
Luigi Logrippio Université du Québec en Outaouais, Canada
Louis Martin LATECE - UQAM, Canada
Hafedh Mili LATECE - UQAM, Canada (Chair)
Charles Petrie Stanford University, USA
Jacques Robert HEC Montréal, Canada

Program Committee Co-chairs

Gilbert Babin HEC Montréal, Canada
Peter Kropf Université de Neuchâtel, Switzerland

Program Committee

Kamel Adi Université du Québec en Outaouais, Canada
Esma Aïmeur Université de Montréal, Canada
Daniel Amyot University of Ottawa, Canada
Gilbert Babin HEC Montréal, Canada
Tony Bailetti Carleton University, Canada
Sarita Bassil Marshall University, USA
Morad Benyoucef University of Ottawa, Canada
Vincenzo D'Andrea University of Trento, Italy
Peter Emmel SAP, Germany
Michael Franz University of California, Irvine, USA
Stéphane Gagnon Université du Québec en Outaouais, Canada

Jaap Gordijn	Vrije Universiteit, The Netherlands
Rüdiger Grimm	University of Koblenz-Landau, Germany
Martin Hepp	Bundeswehr University Munich, Germany
Paul, Hofmann	SAP, USA
Dietmar Jannach	TU-Dortmund, Germany
Gregory Kersten	Concordia University, Canada
Ferhat Khendek	Concordia University, Canada
Peter Kropf	Université de Neuchâtel, Switzerland
Craig, Kuziemsky	University of Ottawa, Canada
Anne-Françoise Le Meur	Université de Lille, France
Luigi Logrippo	Université du Québec en Outaouais, Canada
Simone Ludwig	University of Saskatchewan, Canada
Hafedh Mili	LATECE - UQAM, Canada
Morteza Niktash	Public Works & Government Services, Canada
Liam Peyton	University of Ottawa, Canada
Roy Rada	University of Maryland, USA
Christoph Rensing	Technical University of Darmstadt, Germany
Alain Sandoz	Vauban Technologies, Switzerland
Carlo Simon	Provdadis Hochschule, Germany
Michael Spahn	SAP, Germany
Jun Suzuki	University of Massachusetts, Boston, USA
Thomas Tran	University of Ottawa, Canada
Guy Tremblay	LATECE - UQAM, Canada
Petko Valtchev	LATECE - UQAM, Canada
Marie-Hélène Verrons	Université La Rochelle, France
Michael Weiss	Carleton University, Canada
Yuhong Yan	Concordia University, Canada
Christian Zirpins	University College London, UK

Workshops and Tutorials Committee Chair

Michael Weiss Carleton University, Canada

Industrial Liaison Committee Chair

Liam Peyton University of Ottawa, Canada

Publicity and Sponsorship Committee Co-chairs

Morad Benyoucef University of Ottawa, Canada

Liam Peyton University of Ottawa, Canada

Local Arrangements Committee Chair

Daniel Amyot University of Ottawa, Canada

Reviewers

Abdul Mawlood-Yunis	G.R. Gangadharan	Saeed Behnam
Azalia Shamsaei	Jacques Savoy	Yacine Bouzida
Benjamin Eze	Luc Fabresse	

Table of Contents

Internet-Based Collaborative Work/E-Education

Helping E-Commerce Consumers Make Good Purchase Decisions: A User Reviews-Based Approach	1
<i>Richong Zhang and Thomas T. Tran</i>	
Using Instant Messaging Systems as a Platform for Electronic Voting . . .	12
<i>Anastasia Meletiadou and Rüdiger Grimm</i>	
A Multi-criteria Collaborative Filtering Approach for Research Paper Recommendation in Papyres	25
<i>Amine Naak, Hicham Hage, and Esma Aïmeur</i>	

Industrial Experience

An Ontological Approach to Connecting SOA Implementations	40
<i>Wesley McGregor</i>	
A Non-technical User-Oriented Display Notation for XACML Conditions	53
<i>Bernard Stepien, Amy Felty, and Stan Matwin</i>	
Goal-Driven Development of a Patient Surveillance Application for Improving Patient Safety	65
<i>Saeed Ahmadi Behnam, Daniel Amyot, Alan J. Forster, Liam Peyton, and Azalia Shamsaei</i>	
Global Location-Based Access to Web Applications Using Atom-Based Automatic Update	77
<i>Kulwinder Singh and Dong-Won Park</i>	

Inter-Organizational Processes I

Toward a Framework for Dynamic Service Binding in E-Procurement	89
<i>Maryam Ashoori, Benjamin Eze, Morad Benyoucef, and Liam Peyton</i>	
Integrating Identity Management With Federated Healthcare Data Models	100
<i>Jun Hu and Liam Peyton</i>	

Wrestling With a Paradox: Complexity in Interoperability Standards Making for Healthcare Information Systems 113
Jeff Pittaway and Norm Archer

Open Source and Open Environments

Aligning Goal and Value Models for Information System Design 126
Ananda Edirisuriya and Jelena Zdravkovic

Model-Based Penetration Test Framework for Web Applications Using TTCN-3 141
Pulei Xiong, Bernard Stepien, and Liam Peyton

Impact of Diversity on Open Source Software 155
Hiba Enayat, Steven Muegge, and Stoyan Tanev

Web Search Based on Web Communities Feedback Data 169
Mehdi Adda, Rokia Missaoui, and Petko Valtchev

Inter-Organizational Processes II

Improving Trust and Reputation Modeling in E-Commerce Using Agent Lifetime and Transaction Count 184
Catherine Cormier and Thomas T. Tran

Towards a Methodology for Representing and Classifying Business Processes 196
Hafedh Mili, Abderrahmane Leshob, Eric Lefebvre, Ghislain Lévesque, and Ghizlane El-Boussaidi

Typing for Conflict Detection in Access Control Policies 212
Kamel Adi, Yacine Bouzida, Ikhlass Hattak, Luigi Logrippo, and Serge Mankovskii

Short Research Contributions

Dynamic Pricing in Electronic Commerce Using Neural Network 227
Tapu Kumar Ghose and Thomas T. Tran

TwoStep: An Authentication Method Combining Text and Graphical Passwords 233
P.C. van Oorschot and Tao Wan

Design Principles for E-Government Architectures 240
Alain Sandoz

A Proposed Intelligent Policy-Based Interface for a Mobile eHealth Environment	246
<i>Amir Tavasoli and Norm Archer</i>	

Security and Trust

Verification of Information Flow in Agent-Based Systems	252
<i>Khair Eddin Sabri, Ridha Khedri, and Jason Jaskolka</i>	
A Legal Perspective on Business: Modeling the Impact of Law	267
<i>Sepideh Ghanavati, Alberto Siena, Anna Perini, Daniel Amyot, Liam Peyton, and Angelo Susi</i>	
A Requirement Engineering Framework for Electronic Data Sharing of Health Care Data Between Organizations	279
<i>Xia Liu, Liam Peyton, and Craig Kuziemsky</i>	

Service-Oriented Architecture

An Aspect-Oriented Framework for Business Process Improvement	290
<i>Alireza Pourshahid, Gunter Mussbacher, Daniel Amyot, and Michael Weiss</i>	
Integration Testing of Web Applications and Databases Using TTCN-3	306
<i>Bernard Stepien and Liam Peyton</i>	
A Reference Model for Semantic Peer-to-Peer Networks	319
<i>Abdul-Rahman Mawlood-Yunis, Michael Weiss, and Nicola Santoro</i>	
Author Index	335

Helping E-Commerce Consumers Make Good Purchase Decisions: A User Reviews-Based Approach

Richong Zhang and Thomas Tran

School of Information Technology and Engineering
University of Ottawa
Ottawa, Ontario, Canada
{rzhan025, ttran}@site.uottawa.ca

Abstract. Online product reviews provided by the consumers, who have previously purchased and used some particular products, form a rich source of information for other consumers who would like to study about these products in order to make their purchase decisions. Realizing this great need of consumers, several e-commerce web sites such as Amazon.com offer facilities for consumers to review products and exchange their purchase opinions. Unfortunately, reading through the massive amounts of product reviews available online from many e-communities, forums and newsgroups is not only a tedious task but also an impossible one. Indeed, nowadays consumers need an effective and reliable method to search through those huge sources of information and sort out the most appropriate and helpful product reviews. This paper proposes a model to discover the helpfulness of online product reviews. Product reviews can be analyzed and ranked by our scoring system and those reviews that may help consumers better than others will be found. In addition, we compare our model with a number of machine learning techniques. Our experimental results confirm that our approach is effective in ranking and classifying online product reviews.

Keywords: E-Commerce, Online Product Review, Review Helpfulness, Information Gain, Scoring System.

1 Introduction

Online product review aggregation web sites such as Epinion.com provide consumers with platforms to express and exchange their opinions about products, services and merchants. Consumers who have previously used some specific products or services write reviews of these products and services and rate them by specifying a number of stars. Consumer reviews have become a rich source of information based on which other consumers make purchase decisions. As a matter of fact, online product reviews are showing up as a “new genre” [7], and according to [3] “Online product reviews provided by consumers who previously purchased products have become a major information source for consumers and marketers regarding product quality”.

As consumers try to make good use of online product reviews, several challenging difficulties arise. First of all, with a star rating scheme, consumers can not get the real semantics of reviews. Also, by nature reviews are unstructured and often mix between

reviewers' feelings and opinions. Although search engines are good tools to assist in looking for information, they are not useful in identifying helpful product reviews, since the result set of a query returned by a search engine is often huge and unmanageable. For example, if we input "xbox 360 reviews" in Google, we will receive 47,100,000 web pages, which is clearly impossible for any human to read through. In addition, in a typical online community like Epinion.com, there are usually more than 1000 reviews submitted by different consumers for a specific product. Some review aggregation web sites allow consumers to vote whether or not a review is helpful after they read the review. However, this process takes time far before a really helpful review is discovered. Moreover, the fact that the latest posted reviews are always the least voted ones makes this mechanism inevitably unfair and incomplete. Indeed, consumers need an effective way to classify and rank online product reviews based on the reviews' helpfulness, in order to make good use of this source of information for their purchase decisions.

Our goal is to develop a method that can filter out the most likely helpful reviews for consumers, hence providing reliable information for consumer's decision making. In particular, we propose in this paper an entropy-based model that ranks reviews and returns an ordered list of reviews with their helpfulness estimates. We evaluate the performance of our model using the reviews collected from Amazon.com. The experimental results confirm that our approach outperforms or performs the same as other machine learning methods.

The remainder of this paper is organized as follows: Section 2 discusses related work. Section 3 presents our proposed model in detail. Section 4 describes our experimental evaluation of the model, and Section 5 concludes the paper with future research directions.

2 Related Work

Some researches have been working on sentiment classification, also known as polarity classification, on online product reviews, to distinguish or predict whether consumers like some particular product or not based on their reviews of the product. Hatzivasiloglou et al. proposed a method to predict the semantic orientation of adjectives by a supervised learning algorithm [1]. Turney presented an unsupervised learning algorithm to classify reviews as recommended or not recommended by analyzing the semantic orientation based on mutual information [9]. In [13], the authors proposed a classification approach to separate sentences as positive or negative. In [5], the authors classified movie reviews as positive or negative using several machine learning methods, namely Naive Bayes, Maximum Entropy, and Support Vector Machines (SVM). They also made use of different features such as unigram, bigram, position, and the combination of these features. Their results showed that the unigram presence feature was the most effective and that SVM method performed the best for sentiment classification.

The effect of online product reviews on product sales is also a study area. In [6], the authors discovered that the quality of reviews has positive effect on product sales and that consumers purchase intentions increases with the quantity of product reviews. Hu et al. [3] mentioned that consumers not only considered the review's ratings but also the contextual information like reviewer's reputation. They also found that the impact of online reviews on sales diminishes over time .

Some work has been done in the area of review mining and summarizing. In [14], the authors mined and summarized the movie reviews based on a multi-knowledge approach which includes WordNet, statistical analysis and movie knowledge. Hu and Liu [2] summarized product reviews by mining opinion features.

Evaluating the quality and helpfulness of reviews on web forums is another research domain. Kim et al. [4] delivered a method to automatically assess review helpfulness. They used SVM to train the system and found that the length of reviews, the unigrams, and the product ratings are the most useful features. Weimer et al. [11] proposed an automatic algorithm to assess the quality of reviews in web forums using features such as surface, lexical, syntactic, forum specific, and similarity attributes. In [10], Weimer and Iryna extended the method into three data sets and found that the SVM classification performed very well.

In this paper, we propose a new entropy-based model for scoring the helpfulness of online product reviews. This approach evaluates and ranks online product reviews, helping consumers find the most helpful ones.

3 Proposed Approach

Our work focuses on analyzing reviews and finding high quality and helpful reviews. In this section, we discuss how to estimate the helpfulness and build the helpfulness function.

3.1 Review Helpfulness

Consumers publish their reviews about products online after they finish a transaction. They submit their reviews to web sites like Epinion.com for other potential consumers to read. Also, consumers can vote a review as being “Helpful” or “Not Helpful” after they read the review.

Let C be the set of consumers, P be the set of products, R be the set of reviews, and V be the set of votes, which expresses consumers’ opinions about reviews (possible votes include “Helpful”, “Not Helpful” and “Null”). Thus, we have:

- $C = \{c_1, c_2, c_3, \dots, c_m\}$
- $P = \{p_1, p_2, p_3, \dots, p_n\}$
- $R = \{r_1, r_2, r_3, \dots, r_p\}$
- V is a matrix listed as follows:

$$\begin{aligned}
 & \begin{pmatrix} v_{c_1, r_1} & v_{c_1, r_2} & \dots & v_{c_1, r_p} \\ v_{c_2, r_1} & v_{c_2, r_2} & \dots & v_{c_2, r_p} \\ \vdots & \vdots & \vdots & \vdots \\ v_{c_m, r_1} & v_{c_m, r_2} & \dots & v_{c_m, r_p} \end{pmatrix} \\
 v_{c_i, r_j} &= \begin{cases} \textit{Helpful} & \text{if } c_i \text{ voted } r_j \text{ as Helpful,} \\ \textit{NotHelpful} & \text{if } c_i \text{ voted } r_j \text{ as Not Helpful, or} \\ \textit{Null} & \text{if } c_i \text{ has not voted for } r_j. \end{cases} \quad (1)
 \end{aligned}$$

Definition 1. Review helpfulness is the perception that a review $r \in R$ can be used to assist consumers to understand a product $p \in P$. For a particular review $r_x \in R$, its helpfulness is calculated as the ratio of the number of consumers who have voted r_x as being “Helpful” to the total number of consumers who have voted for r_x .

Let the set of all the “Helpful” votes about review r_x be denoted by h_x . Let the set of all “Not Helpful” votes about review r_x be denoted by \bar{h}_x . According to the above definition, the helpfulness of review r_x is:

$$\frac{|h_x|}{|\bar{h}_x| + |h_x|} \quad (2)$$

Reviews can be rated by consumers after they read them. We predefine 0.60 as a threshold for helpfulness. If the review’s helpfulness is greater than 0.60, we say that it is helpful. An online review consists of words, which include opinion words, product features, product parts, and other words. The importance of each word to the helpfulness of the review can be calculated from a training data which contains the vote information provided by consumers. As described below, we use an entropy-based method to rank reviews.

3.2 Entropy and Information Gain

In the work of Pang et al. [5], the authors reported that the best result was obtained by using the boolean values of unigram features. We use bag of words model to represent text and build our language model. Each feature is a non-stop stemmed word and the value of this feature is a boolean value of the occurrence of the word on the review.

We introduce the Shannon’s information entropy concept [8] to measure the amount of information in reviews. The entropy can be extended as follows:

Let $S = \{s_1, s_2, \dots, s_q\}$ be the set of categories in the review space. The expected information needed to classify a review is:

$$H(S) = - \sum_{i=1}^q P_r(s_i) \log P_r(s_i) \quad (3)$$

The average amount of information contributed by a term t in a class s_i will be:

$$H(S|t) = - \sum_{i=1}^q P_r(s_i|t) \log P_r(s_i|t) \quad (4)$$

Information Gain is derived from entropy and can be understood as the expected entropy reduction by knowing the existence of a term t .

$$G(t) = H(S) - H(S|t) \quad (5)$$

Information gain is often employed as a term goodness criterion in the field of machine learning [12] and often used as a feature selection method in text classification. In [12], the information gain of term t is extended and defined as follows:

$$\begin{aligned}
G(t) = & - \sum_{i=1}^q P_r(s_i) \log P_r(s_i) + \\
& P_r(t) \sum_{i=1}^q P_r(s_i|t) \log P_r(s_i|t) + \\
& P_r(\bar{t}) \sum_{i=1}^q P_r(s_i|\bar{t}) \log P_r(s_i|\bar{t})
\end{aligned} \tag{6}$$

In the above formulas:

- $P_r(s_i)$ is the probability of documents in category s_i among all documents.
- $P_r(t)$ is the probability of documents which contain term t among all documents.
- $P_r(\bar{t})$ is the probability of documents which do not contain term t among all documents.
- $P_r(s_i|t)$ is the probability of documents which contain term t and which belong to category s_i , out of all documents which contain t .
- $P_r(s_i|\bar{t})$ is the probability of documents which do not contain term t and which belong to category s_i , out of all documents which do not contain t .

The above formula calculates the reduction of entropy by knowing the occurrence of a specified term. It considers not only the term's occurrence, but also the term's non-occurrence. This value can somehow indicate the term's contribution and predicting ability. If a word has higher information gain, it has more contribution for classifying. For binary classification, this value can be used to measure the amount of contribution of this a term to a class.

In our case, only two categories, "Helpful" and "Not Helpful", will be considered. Let s_1 be "Not Helpful" and s_2 be "Helpful". In order to provide the difference of prediction ability for these two categories, we introduce the following change: if $P(s_1|t) < P(s_2|t)$ then $Gain(t) = G(t)$; otherwise, $Gain(t) = -G(t)$.

So the Gain value of term t in our model is calculated as follows:

$$Gain(t) = \begin{cases} G(t) & \text{if } P(s_1|t) < P(s_2|t), \\ -G(t) & \text{otherwise.} \end{cases} \tag{7}$$

Where s_1 is the category of "Not Helpful", s_2 is the category of "Helpful".

Thus, the importance and the prediction ability of words can be calculated by equation (7). Table 1 shows an example of information gain values. The second and the third columns show the occurrence times of the specific term in the "Helpful" and "Not Helpful" domains, respectively.

3.3 Prediction Computation

From the discussion in the above subsection, the Gain value can represent words' ability to correctly predict a document belonging to the "Helpful" or "Not Helpful" category. So, the summarization of the Gain values of all words in a review indicates the review's helpfulness. In our approach, a review's content (words) will be analyzed and the Gain

Table 1. Information Gain Value Example

term	Not Helpful	Helpful	Information Gain
nuvi	55	174	0.086522889
bluetooth	11	93	0.072981165
mount	13	96	0.071278275
screen	33	118	0.055793201
crash	10	2	-0.004942425
uninstal	7	0	-0.008155578
minimum	10	0	-0.011693703

value will be calculated for each word (not including stop words) in the review. To calculate the helpfulness of a review r_i , we propose the score calculation equation as follows:

$$Score(r_i) = \sum_{j=1}^N Gain(t_j) * f(r_i, t_j) \quad (8)$$

where $Gain(t_j)$ is the Gain value of the j^{th} stemmed word, N is the total number of stemmed words in review r_i .

$$f(r_i, t_j) = \begin{cases} 1 & \text{if term } t_j \text{ occurs in } r_i, \text{ or} \\ 0 & \text{if term } t_j \text{ does not occur in } r_i. \end{cases} \quad (9)$$

Equation (8) can be seen as the total helpfulness information delivered by review r_i . This equation can be used as a model to predict the helpfulness. All the score values of reviews $r_i \in R$ will be calculated. As a result, tuples of the form $\langle r_i, Score(r_i) \rangle$ will be returned by this approach (where r_i is an online product review and $Score(r_i)$ is review r_i 's helpfulness value). Finally, online product reviews will be ranked based on their corresponding $Score(r_i)$ values. Reviews with higher $Score$ values will be more helpful than others. With a set \mathbf{T} of training reviews of a specific product category, and a set \mathbf{T}' of test reviews, the helpfulness prediction process will be as follows:

1. Find the $Gain$ values for every non-stop word from \mathbf{T} .
2. Calculate the $Score$ value for every review in \mathbf{T}' by equation (8).
3. Sort \mathbf{T}' in descending order based on their $Score$ values.

4 Experimental Evaluation

In this section we first define some necessary measures. Then, we describe the data set and the experiment steps. Finally, we analyze the experimental results and evaluate the performance of our approach.

4.1 Measures

To evaluate the performance of our model, precision and recall rates are used. Precision and recall are commonly used in evaluating information retrieval systems. Precision is

defined as the ratio of retrieved helpful reviews to the total number of reviews retrieved. Recall is defined as the ratio of the number of retrieved helpful reviews to the total number of helpful reviews.

$$Precision = \frac{\text{Helpful reviews found}}{\text{Total reviews found}} \quad (10)$$

$$Recall = \frac{\text{Helpful reviews found}}{\text{Total helpful reviews}} \quad (11)$$

F-Measure is defined as the harmonic mean of the above two measures and is calculated by

$$F = \frac{2 * Precision * Recall}{Precision + Recall} \quad (12)$$

We also apply Pearson's correlation coefficient between ranks to evaluate the ranking performance of our model. This method will assess the relation between predicted ranks and real ranks.

4.2 Data Set

We crawled 9955 GPS and MP3 Player reviews from Amazon.com. Each of these reviews has been evaluated by at least 4 consumers as helpful or not helpful. We define that if the helpfulness of a review (the percentage of helpful votes) is greater than 60%, the review will be marked as helpful; otherwise, it is not helpful. 720 GPS reviews and 800 MP3 Player reviews are randomly selected to do the experiment.

After the parsing and stemming to all the training reviews, a document term matrix is returned associated with the *Gain* value of each stemmed word. We use 10-fold cross validation to evaluate our approach. Reviews are randomly divided in to 10 equal-sized folds. 9 folds of reviews are used for training the model and one fold is used as test data.

4.3 Results and Analysis

In order to classify the sorted reviews into "Helpful" and "Not Helpful" categories, a Helpfulness threshold is need. We define the Helpfulness threshold to be the $|Helpful|^{th}$ sorted review's corresponding *Score* value, where $|Helpful|$ is the number of helpful reviews in the training set. We use this threshold to analyze the classification precision and recall of our model. Figure 1 and 2 show the distribution of reviews' score values of the experimental result. Most of the helpful data and not helpful data concentrate on the two end of the score value space. Helpful data will have bigger score and not helpful data will have smaller score. This distribution highly indicates that the *Score* function can model the helpfulness of online product reviews. Therefore, with the ranking of scores, most of helpful reviews can be retrieved on the top of the sorted reviews' list.

Table 2 shows the classification performance of our model. The classification precision for GPS reviews is 77.7% for helpful reviews, and 77% for not helpful reviews.

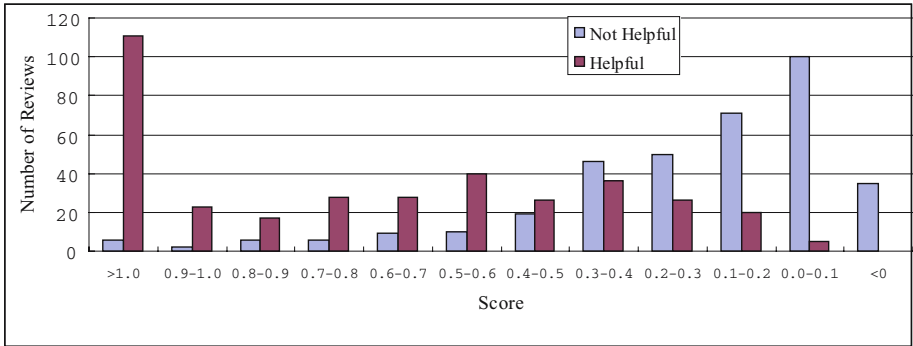


Fig. 1. Distribution of Reviews' Score

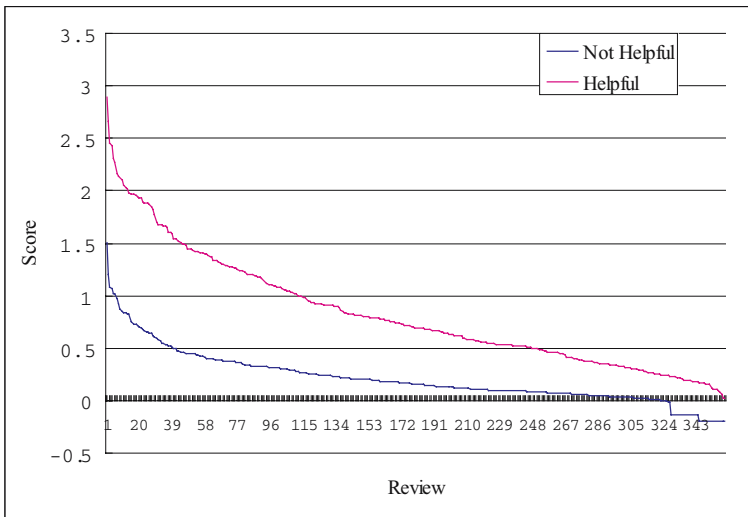


Fig. 2. Score Values of Reviews

Among the MP3 Player reviews, the precision is 69.7% for not helpful reviews and 72% for helpful reviews. Table 3 is the result of 10-fold cross validation by our model and other classification methods for GPS reviews. We compare the precision, recall, and F-measure for Naive Bayes, Decision Tree, SMO and our entropy-based model. The result listed in Table 3 reveals that our model performs better or the same as other machine learning classification methods.

Table 4 reports the ranking quality tests of rank correlation coefficient. The rank correlation coefficient between the helpfulness ranking generated by our model and the helpfulness ranking voted by consumers is 0.58 for GPS and 0.51 for MP3 Player. This shows that the predicted helpfulness ranking of our model and the original helpfulness ranking of consumers are in good correlation.

Table 2. Precision, Recall and F-Measure (10-fold cross-validation)

	Precision		Recall		F-Measure	
	MP3 Player	GPS	MP3 Player	GPS	MP3 Player	GPS
Not Helpful	69.7%	77%	73.5%	78.1%	71.5%	77.5%
Helpful	72%	77.7%	68%	76.7%	69.9%	77.2%

Table 3. Performance of various classification methods and our model (10-fold cross-validation)

	Precision		Recall		F-measure	
	Helpful	Not Helpful	Helpful	Not Helpful	Helpful	Not Helpful
Naive Bayes	0.747	0.768	0.778	0.736	0.762	0.752
SMO	0.796	0.749	0.728	0.814	0.761	0.78
Decision Tree	0.77	0.714	0.681	0.797	0.723	0.753
Our Model	0.777	0.77	0.767	0.781	0.772	0.775

4.4 Discussion

The model discussed in this paper analyzes the gain values of each word from the training set. Information gain is assigned plus or minus sign based on the helpfulness. With the “Gain”, features and their corresponding importance can be discovered. This model performs quite well for the data set of Amazon.com based on the result analysis above.

Rank correlation coefficient is one of the most common methods to compare two different rankings on the same set of items in statistics. If the correlation between the two rankings is perfect, the value is 1. If the two rankings are totally diverse of each other, the value is -1. Table 4 shows the average correlation coefficient values for the ranking manually voted by consumers and the ranking predicted by our model on two product categories, MP3 players and GPS. The correlation coefficient found is 0.58 for GPS reviews and 0.51 for MP3 Player reviews, which demonstrates a good correlation between the two rankings.

Table 4. Evaluation of performance of our model ranking reviews of GPS and MP3 Players (using 10-fold cross-validation)

Collection	Pearson correlation
GPS	0.58
MP3 Player	0.51

Also, our experimental results show that the classification performance of our approach is better or close to other commonly used classification methods. Our approach outperforms the Naive Bayes and Decision Tree methods. In comparison with the SMO method, our approach is about 1% lower for the F-measure of “Not Helpful” reviews, and 1% higher for the “Helpful” reviews.

5 Conclusion and Future Work

This paper proposes an approach to model the helpfulness of online product reviews. Using this model, online product reviews can be classified and ranked based on their score values. Thus, the proposed approach provides an effective means for consumers to find the most helpful reviews in order to make their purchase decisions. We made use of the entropy and information gain concept and revised them to create a suitable model. In comparison with other machine learning classification methods, our model is simpler, easier to understand and implement. The time complexity of our model is $O(D * W)$, where D is the number of reviews in the training set and W is the number of non-stop words in the training set. The proposed model can classify and rank reviews quickly. The experimental results show that our method can reach the precision of 77% for GPS reviews based on a predefined threshold. Also, with the 10-fold cross validation evaluation, our model can get the rank correlation coefficient of 0.58 for GPS reviews and 0.51 for MP3 Player reviews.

Reviews from other product categories and much bigger review sets will be observed in the future. Future research will also take into consideration other factors, which may affect the quality of a review such as when the review was published, how consumers rated the product, and the number of features that have been mentioned in the review. Also, in this paper we assumed that all consumers have the same preferences for online reviews and did not consider the difference of individuals. In the future, in order to improve the accuracy of personalization, the similarity between consumers will be considered. We note that our model can be introduced to a search engine to enhance the helpfulness of search results. We are collecting more data to examine these possible research directions.

References

1. Hatzivassiloglou, V., McKeown, K.R.: Predicting the Semantic Orientation of Adjectives. In: Proceedings of the Eighth Conference on European Chapter of the Association for Computational Linguistics, pp. 174–181. Association for Computational Linguistics, Morristown (1997)
2. Hu, M., Liu, B.: Mining and Summarizing Customer Reviews. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2004), pp. 168–177. ACM Press, New York (2004)
3. Hu, N., Liu, L., Zhang, J.: Do Online Reviews Affect Product Sales? The Role of Reviewer Characteristics and Temporal Effects. In: Information Technology and Management (2008)
4. Kim, S.M., Pantel, P., Chklovski, T., Pennacchiotti, M.: Automatically Assessing Review Helpfulness. In: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, pp. 423–430. Association for Computational Linguistics (2006)
5. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: Sentiment Classification Using Machine Learning Techniques. In: Proceedings of the ACL 2002 conference on Empirical methods in natural language processing (EMNLP 2002), pp. 79–86. Association for Computational Linguistics, Morristown (2002)
6. Park, D.H., Lee, J., Han, I.: The Effect of On-Line Consumer Reviews on Consumer Purchasing Intention: The Moderating Role of Involvement. *Int. J. Electronic Commerce* 11(4), 125–148 (2007)

7. Pollach, I.: Electronic Word of Mouth: A Genre Analysis of Product Reviews on Consumer Opinion Web Sites. In: HICSS, vol. 3, pp. 1530–1605. IEEE Computer Society, Los Alamitos (2006)
8. Shannon, C.E.: A Mathematical Theory of Communication. *SIGMOBILE Mob. Comput. Commun. Rev.* 5(1), 3–55 (2001)
9. Turney, P.: Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews,
<http://citeseer.ist.psu.edu/turney02thumbs.html>
10. Weimer, M., Gurevych, I.: Predicting the Perceived Quality of Web Forum Posts. In: Proceedings of the Conference on Recent Advances in Natural Language Processing, RANLP 2007 (2007)
11. Weimer, M., Gurevych, I., Mühlhäuser, M.: Automatically Assessing the Post Quality in Online Discussions on Software. In: Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, pp. 125–128. Association for Computational Linguistics (2007)
12. Yang, Y., Pedersen, J.O.: A Comparative Study on Feature Selection in Text Categorization. In: Proceedings of the Fourteenth International Conference on Machine Learning (ICML 1997), pp. 412–420. Morgan Kaufmann Publishers Inc., San Francisco (1997)
13. Yu, H., Hatzivassiloglou, V.: Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying the Polarity of Opinion sentences. In: Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing, pp. 129–136. Association for Computational Linguistics, Morristown (2003)
14. Zhuang, L., Jing, F., Zhu, X.Y.: Movie Review Mining and Summarization. In: Proceedings of the 15th ACM International Conference on Information and Knowledge Management (CIKM 2006), pp. 43–50. ACM Press, New York (2006)

Using Instant Messaging Systems as a Platform for Electronic Voting

Anastasia Meletiadou and Rüdiger Grimm

University Koblenz-Landau
Universitaetstrasse 1, 56070 Koblenz, Germany
{nancy,grimm}@uni-koblenz.de

Abstract. Many Instant Messaging (IM) systems like Skype or Spark offer extended services such as file sharing, VoIP, or a shared whiteboard. As the name suggests, IM applications are predominantly used for spontaneous text-based communication for private or business purposes. In this paper we explore their potential to serve as platforms for secure collaborative applications like electronic contract negotiation, e-payment or electronic voting. Such applications have to deal with challenges like time constraints (“instant” communication is desired), integration of media channels and the absence of one unifying “sphere of control” covering all participants. In this paper, we address these challenges by discussing one particular secure collaborative application: secure decision processes for small groups. We provide the following contributions: (1) we define three varying scenarios and corresponding security requirements (2) we present an IM-based architecture implementing these scenarios, including a Video-based authentication mechanism, and (3) we discuss potential attack patterns.

Keywords: Instant messaging, collaboration, electronic voting, security.

1 Introduction

Nowadays companies require for new flexible communication channels to support distributed projects and collaborations. The research area Computer Supported Cooperative Work (CSCW) deals with approaches for collaborative activities and their coordination that can be supported by computer systems. One subcategory of CSCW tools are Instant Messaging Systems (IMS). Strictly speaking, instant *messaging* systems support only text-based communication (chat). However, we include advanced services, such as voice-based communication (Voice over IP), video over IP and file sharing, as well.

In the research project underlying this paper we aim to explore the potential of Instant Messaging Systems to serve as platforms for secure collaborative application. Consequently, in this paper we (1) focus on decision processes in small groups as one specific type of such collaborative applications and (2) analyse if and how such processes could be supported by applications based on Instant Messaging Systems. Every group process involves two parts: a discussion and a subsequent voting.

The research methodology which we used is Design Research [1], which consists of five steps: Awareness of Problem, Suggestion, Development, Evaluation, and Conclusion. In this paper we raise the awareness of the problem (spontaneous and non-repudiate decision processes, section 2) and we suggest an architecture as a solution and show the initial stage of an implementation (sections 3 and 4). The evaluation phase is beyond the scope of this paper.

2 Description of Decision Scenarios

In the following discussion we will discuss a set of related scenarios for decision processes in small groups (about three to twenty persons). These scenarios are selected because they cover the different impacts of voting in a group decision. Depending on the concrete scenario, we have to enforce *varying* rules. For instance, a discussion (to collect suggestions for a ballot) can be performed by name and subsequent voting can be performed by name or anonymously. Consequently, we have to implement these different security goals and requirements in the underlying software platform.

2.1 Scenario: No Discussion, Secret Election

The first scenario is a decision process without discussion, but with a secret ballot. Such a process is used, for instance, in universities to decide on the membership in committees and boards. Usually, such elections are organised by a small election commission and take place on the premises of the university. There is a list of candidates (the ballot paper), which is created by the election commission and, depending on the situation, there might be a polling booth. During the process certain rules and regulations for the election apply, e.g., the election should be free, equal and secret [2][3]. These rules are the basis for the following security requirements:

The *anonymity* is ensured by the trust model of the election authority. Elections in universities are organised similarly to political elections (in Germany). The election authority ensures that only eligible voter cast a vote. To this end, the voter shows his identity card (authentication), which will be checked against the electoral register (proof of eligibility). The voter receives his ballot paper; he secretly makes his choice in a polling booth and casts his vote. Hence, there is no connection between his name (his identity) and his vote.

The *non-repudiation* property appears in two forms: 1) a voter can not repudiate *that* he has voted (the participation can be proved) and 2) a participant can not repudiate *how* he has voted (which, of course, must be prevented in order to ensure a secret election!). In the first case the participation can be proven by confirmation by other present participants or by ticking off the corresponding entry in the electoral register (when the authority verifies the voter's right to vote and then hands out a ballot).

Another requirement is the *integrity of the results*. Usually, the voter is allowed to observe the whole voting procedure, if he requires. Even if he does not want to observe the procedure himself, he can (decide to) trust the election authority. The authority is trustworthy, because the pool workers have different interests for the election, so it is very difficult to manipulate all of them. If someone, nevertheless, doubts the result of

an election, he has the option to raise an objection and, depending on the particular scenario and effort required, the counting or even the whole voting procedure can be repeated.

Finally, the election authority counts the votes and publishes the result.

2.2 Scenario: No Discussion and Election by Name

This second scenario is a special case of the next scenario “discussion and election by name” (see below). Accordingly, the applied requirements remain the same, only the ballot paper is a result of a previous process and not a result of the current decision process.

2.3 Scenario: Discussion and Election by Name

This third scenario is a decision process with discussion and subsequent named voting. Such a process occurs in universities, for example when an extension of the curriculum has to be decided, for instance the introduction of a new lecture. The members of a committee discuss the alternatives, their advantages and drawbacks and then vote for their preferred alternative.

The requirements for the subsequent voting are as follows: *Anonymity* is not required; every member should know the available choices and opinions of the other members. Every voter can determine the result by himself, because he can see and count the votes. Hence, the *non-repudiation* property and the *integrity of the results* are achieved. Similar to the preceding scenario, someone has to check the eligibility and authenticity of the members. In practice, the authentication of voter occurs without the use of identity cards or other identification mechanism. In such small groups the participants know each other by personal acquaintance and through the social interaction in day-to-day work.

During the decision process a transcript is created and the results are published.

2.4 Scenario: Discussion by Name and Secret Voting

This scenario is a combination of the two preceding scenarios (2.1 and 2.3) combining open discussion and secret voting. Such processes occur in universities for example when a selection committee has to decide on applicants applying for an open position.

The members of the committee openly discuss the applicants and then make a decision by secret election. In this scenario the *anonymity* is required only for the second part; the first part is a named discussion (i.e., identities of participants are known) which also produces a list of available choices for the ballot paper. Subsequently, the secret voting is carried out. The chairman of the committee collects and counts all votes; he then announces the results. In one variant of this procedure, the chairman collects the votes, but performs a public voting, such that every participant can trace the calculation of the result.

The requirement of the *non-repudiation* property is the same as in the first scenario. The only difference is that it is possible to gather information on individual opinions and derive certain voting preferences.

Again, the authentication of voter occurs without identity cards or other identification mechanism. If a participant has doubts in the results or their integrity, then the counting can be repeated.

During such a decision process, usually a transcript is created and published later on.

3 Instant Messaging as a Platform for E-Voting

To technically implement such processes we use an IM system as a foundation and extend it with application-specific plug-ins. The idea is to find a way to support collaborating processes, like a decision process with subsequent voting, even in situations where the participants are distributed across various locations. With this approach implemented with IM technologies, we want to support the same security requirements and sequence of decision events, just like in a conventional approach where the decision is made on-site in a face-to-face meeting.

3.1 Software Architecture of the Extended IM System

Figure 1 illustrates the communication between two IM clients.

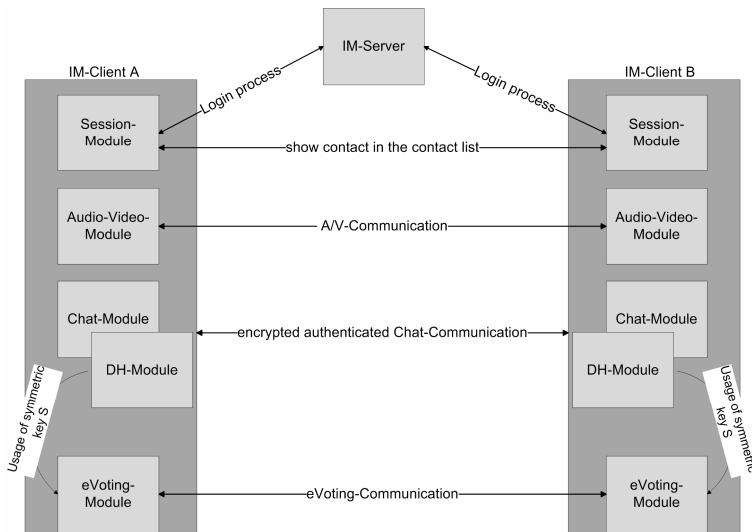


Fig. 1. Architecture of the extended IM system

Our extensions (modules) implement the follow functions: an authentication mechanism for the decision process (*DH-module*), the interactive creation of a ballot paper and the implementation of the vote protocol, i.e., carrying out the vote and determining the results (*eVoting-module*)

As a foundation for our implementation we selected the Spark IM Client [5] and the Openfire IM Server [6]. The main reasons for this decision are that Spark and

Openfire are open source software, that there is a large developer community and that they support the standard protocol Jabber/XMPP. The usage of the XMPP protocol was important because this allows us to define own customized messages types for the IM communication, which we use to transport application specific data. For example, this is required for the authentication with an adapted Diffie-Hellman mechanism and for the interactive implementation of the casting of ballots.

3.2 Overview of the Voting Procedure

In order to give more details about our approach, we will now explain the overall procedure including the communication between the participants. To define the “protocol” for the interaction between the participants, we have to define rules for this communication. There are two kinds of rules:

First, there are *rules for the admission to the group* – These are rules for the proof of identification, for the authentication and for announcing the new participant in the group.

Second, after admission has been granted, there are *rules for the interaction between the members* – These rules apply to the actual communication and cooperation between the participants.

Depending on the IM system the communication between clients is implemented differently. Some systems transfer the whole communication indirectly via the IM server. Hence, there is a central point of control. Other systems direct only the registration and authentication via the IM server, but implement the main communication directly between the clients.

We will now give an overview of the overall procedure (see figure 2, left part) and consider one part of it, the authentication mechanism, in more detail (see figure 2, right part).

First, the *authentication mechanism of the IM System* is applied, i.e., during the login process the user name and password, as entered by the user, are compared against the profile stored in the user database on the authentication server of the IM system. This mechanism provides some protection against usage of accounts by unauthorized persons – once the right user has set up his account. However, during registration of a new user accounts, this procedure does not enforce a mapping of the account to the *right* real-life person. For instance, in many IM systems a new user can freely chose any user name, as long as it has not been taken before. Since a mapping of user accounts to correct real-life persons is important in secure collaborative applications (such as our voting scenarios) we have to find a better solution. A potential solution would be the establishment of a third party that checks the mapping between an identity and the chosen account, for instance by means of an identity card. However, this concept requires considerable effort for each new user and is realistic only if all participants belong to the same institution. In contrast, we want to provide an approach which (1) provides a security level comparable to person-to-person meetings and (2) does not require complicated registration processes or the existence of a central infrastructure or administration (i.e., a PKI).

This is especially helpful in scenarios where the members of the decision board belong to different organisations and are not covered by the same organisational infrastructure. In a real-life committee meeting the “authentication mechanism” is given

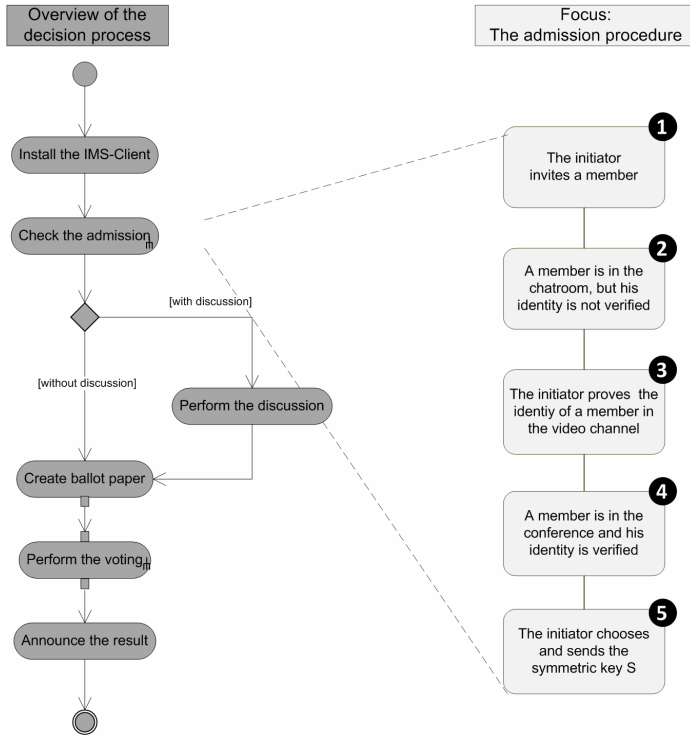


Fig. 2. Overview of the voting process

by the personal acquaintance between the members. Every member either recognizes other participants or at least there exists an indirect connection. The goal of our approach is to provide a technical mechanism, which reproduces the security by personal recognition as good as possible.

This is especially helpful in scenarios where the members of the decision board belong to different organisations and are not covered by the same organisational infrastructure. In a real-life committee meeting the “authentication mechanism” is given by the personal acquaintance between the members. Every member either recognizes other participants or at least there exists an indirect connection. The goal of our approach is to provide a technical mechanism, which reproduces the security by personal recognition as good as possible.

Consequently, we add *extensions which provide a second authentication mechanism*. The mechanism uses a combination of (1) recognition of other collaboration partners in a video picture and (2) protection of the communication channel with an adapted Diffie-Hellman (DH) mechanism [7]. The integrity of the Diffie-Hellman-protected connection is checked by reading out and comparing a hash value.

Zimmermann uses a similar principle [8]. However, his approach is restricted to video as a communication medium and supports bilateral communication only (no group collaboration possible).

Here, our approach provides improvements in both counts: (1) We extend the secure communication to support application-specific protocols (e.g., e-voting communication) and (2) we support group communication with more than two participants.

Our approach works as follow: The members of a group arrange a meeting to make a decision. Figure 2 shows an overview of the process. One person, for example the chairman, takes the role of the initiator of the whole process. In the following we call this initiator “person A”. He invites the other members to a common chatroom. This is step ❶ in the activity diagram in figure 2. The corresponding user interface of the application (as seen by A) is shown in figure 3.

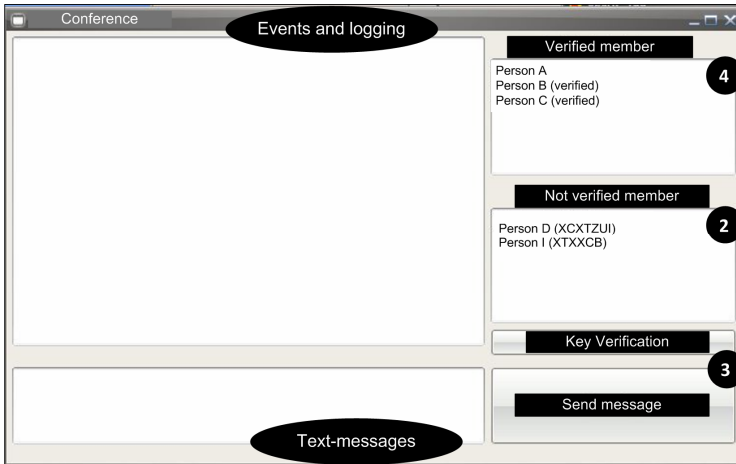


Fig. 3. The User Interface, as seen by A, during authentication

Every member who enters the chatroom has to perform a DH key-agreement with the initiator, person A (Step ❷ in figures 2 and 3). As an example consider the situation shown in Figure 4. For these four persons meeting (with participants A, B, C, and D), three DH-Keys K_{AB} , K_{AC} , and K_{AD} would be created. Subsequently, the initiator A knows all these keys, the other participants only know their bilateral key. For instance, B only knows the key K_{AB} .

After this key agreement, the initiator A verifies the identities of B, C and D by recognition in the video channel (Step ❸ in figures 2 and 3). For this, every participant reads out a hash value of the bilateral keys (i.e. K_{AB} for the connection between A and B). During this reading of hash values the initiator has to check two things: (1) if the person in the video is the authentic partner (correspondence between the account name and the real person in the video picture) and (2) if the connection ensures data integrity (equality between the hash value as known to the initiator and the corresponding hash value as know to the person at the other end of the connection). If these checks are completed successful, then (1) the person is regarded as authenticated and (2) the connection is regarded as secure (Step ❹ in figures 2 and 3). Finally, the initiator A creates a symmetric key S and distributes it to the authenticated members (Step ❺ in figure 2). The symmetric key S is used to encrypt further communication between the (now authenticated) partners, including protocols of the integrated application, e.g., e-voting data.

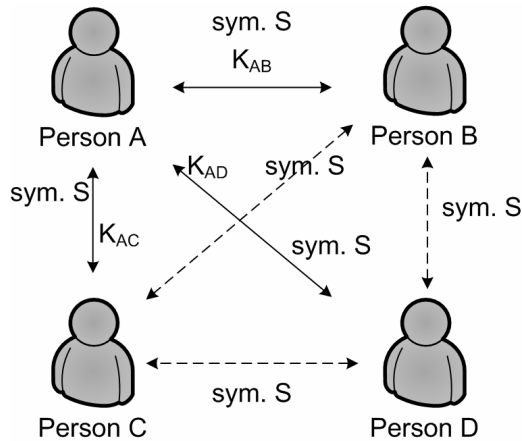


Fig. 4. Key exchange during the authentication

3.3 Attacks and Countermeasures

In this section we present potential attacks and the corresponding solutions.

3.3.1 The Initiator is not Trustworthy

If the initiator (authority, person A) is not trustworthy, then there are two possible manipulations: 1) the misconducting initiator allows a person, which was not authenticated or recognized in the video channel, access to the group (deliberate Man-in-the-Middle attack). 2) The initiator infringes his duty of care and permits the entry of a person, which is not entitled to vote. In other words, the authentication and recognition is successful, but with the “false” person.

A manipulation can be discovered in the course of discussion or a decision, for instance if a participant is irritated by the behaviour or statements of another member. In this case the communication is interrupted and a new decision process is started.

3.3.2 A Member Betrays the Shared Symmetric Key S

Another attack is the passing on of the symmetric key S. For example, member D could pass on the key to an attacker called E. In such a case, the attacker E is able to eavesdrop and hear the communication, but he is not able to manipulate the communication or the decision process. A manipulation of a connection is possible as a Man-in-the-Middle attack (see below) or if a participant leaves his position (and the corresponding right to vote) to a third person. Such a manipulation is difficult to detect owing to the member’s collaboration.

One way to detect such manipulations nevertheless, is an extension of the proposed authentication process: If a participant doubts the identity and the answers of another member of the group (in the chat channel), he requests the repetition of the authentication process (with recognition in the video channel). Obviously, a new symmetric key has to be generated and distributed, if an attack is discovered this way.

3.3.3 Man-in-the-Middle-Attack (One Connection Is Manipulated)

In section 2.3.1 we discussed a deliberate Man-in-the-Middle attack, where an insider, who is member of the group, provides information to an outsider, who becomes the Man-in-the-Middle. There is, however, another type of Man-in-the-Middle attack, which works without insider support, e.g., by infiltration of a participant's computer with malware. In such a case, the attacker manipulates the connection between two members, such that the participants see different content in their chat window, although all members are in the same chat room. One aim of such an attack (against the authenticity and integrity) could be to manipulate opinions and statements. Detection is possible by consistency check, for instance through comparison of related statements. If such a case is detected, the communication should be interrupted.

In our implementation we use a signature for each message to ensure the authenticity and integrity of the communication. To this end, we use public and secret keys which are created during the DH key agreement. Another option might be the use of a central public key infrastructure (PKI). However, we want to avoid such an architecture, since this is against our goals, as it hinders spontaneous and immediate communication.

4 E-Voting

Our approach implements decision processes in small groups, including a discussions part and an e-voting part.

In principle, in such groups all participants have equal rights. That means that every participant can potentially become the initiator of the decision process. Moreover, every participant should be able to comprehend the e-voting method and to reproduce the e-voting results by counting himself. Hence, Secret-Sharing protocols [9] or Mix-Method protocols [10] are suitable for e-voting in small groups.

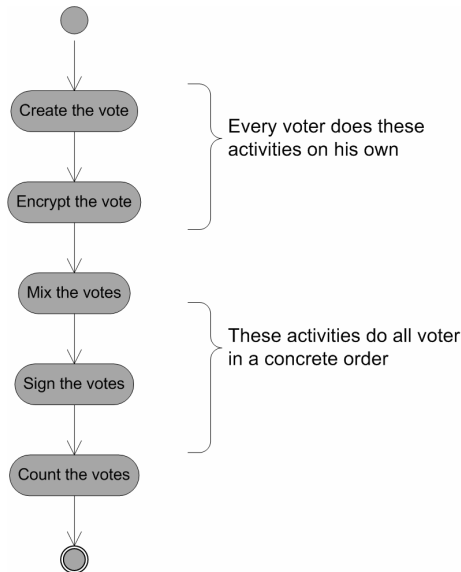


Fig. 5. Overview of the voting protocol

In our approach we use an extension named “evote” [11] of the Mix-Method protocol, which in turn is based on the research work of [10]. For the limited context of this paper, we will focus on the main ideas of the Mix-Method protocol, which works as follows (see figure 5):

4.1 Creation and Encryption of votes

At the beginning a sequence is defined, which determines the order in which participants will exchange messages. The particular order is arbitrary, but has to be known to all participants. For example, we could use an alphabetical order (ABCD). Every voter makes his choice (create the vote, in figure 5) and encrypts his vote with all public keys of the other members (encrypt the vote, in figure 5). For that, he combines his own vote with a random number and then encrypts it with all public keys in the reserve order (of the sequence defined earlier). In our example this would be DCBA.

$$RES(X) = Pub_A (Pub_B (Pub_C (Pub_D (V_x, R_x) . \quad (1)$$

The result RES will be encrypted in a second round with all public keys combined with more random numbers. These random numbers will later serve to enable each voter to check whether, after some message exchange and mixing, his vote still is in the set of collected votes. To prepare for these checks, every member X encrypts his result Res(X) as follows:

$$V(X) = Pub_A (R_{XA}, Pub_B (R_{XB}, Pub_C (R_{XC}, Pub_D (R_{XD}, RES(X)))) \quad (2)$$

4.2 Mixing of Votes

After all votes have been prepared in this way, they will be sent along a chain of participants, which is given by the sequence defined at the beginning of the procedure (ABCD). During the preceding step each vote was encrypted with keys *in reverse order* (DCBA), i.e., first encrypted with D’s key and so on (see equation 2). Hence, each participant along the sequence ABCD can process and remove the outermost encryption. This works as follows:

First A receives the votes from all participants. He decrypts the messages with his private key. He then ensures that his vote is still in the pool, by checking if one of the votes in the pool contains the random number that he generated earlier while preparing his vote (equation 2). He then mixes all votes and sends them on to the next member in the sequence, i.e., person B (*mix the votes, in figure 5*).

Person B receives four votes, with the outermost encryption removed by A:

$$V(X) = Pub_B (R_{XB}, Pub_C (R_{XC}, Pub_D (R_{XD}, RES(X))) \quad (3)$$

B performs the same as A (decrypt-check-mix) and sends the votes on to the next station. This is repeated until all members including D have completed the decrypt-check-mix procedure.

4.3 Signing of Votes

In the next phase the votes are sent along the sequence of participants a second time to allow each participant to check that his vote is still present. Initially, person A receives four votes of the form:

$$V(X) = \text{RES}(X) = \text{Pub}_A (\text{Pub}_B (\text{Pub}_C (\text{Pub}_D (V_x, R_x))) \quad (4)$$

Since the votes have been mixed multiple times, A does not know which vote is which. However, he can check whether his random number R_A is present. Hence, A decrypts the votes and gets four votes of the form:

$$V(X) = \text{Pub}_B (\text{Pub}_C (\text{Pub}_D (V_x, R_x)) \quad (5)$$

He then checks again, if his vote is present and signs all votes and sends these to B as the next member in the sequence (for processing) and to all other members (to enable them to observe and check the procedure). Hence, A signs all four $V(X)$ and combines them into one message. By sending this combined message to all participants he acknowledges that his vote is in one of these votes:

$$\text{allvote} := \text{sig}_A V(B), \text{sig}_A V(A), \text{sig}_A V(D), \text{sig}_A V(C) \quad (6)$$

The next one in the sequence, B repeats this procedure producing:

$$V(X) = \text{Pub}_C (\text{Pub}_D (V_x, R_x)) \quad (7)$$

$$\text{allvote} := \text{sig}_B V(B), \text{sig}_B V(A), \text{sig}_B V(D), \text{sig}_B V(C) \quad (8)$$

Finally, the last member D receives the four votes, which he checks, signs and sends to all members. These votes which are sent to all participants have the form:

$$V(X) = (V_x, R_x) \quad (9)$$

4.4 Counting of Votes

Now each member receives all decrypted votes. By checking for his random number, which is only known to him, he can ensure that his own vote is part of the final result and will be counted. In consequence, everyone can count and reproduce the result (*count the votes, in figure 5*).

4.5 Suitability of the Mix-Method Protocol for Our Scenarios

For our approach and scenarios which were explained in the sections 2 and 3 we need a protocol which fulfils the following requirements: It is suitable for small groups; it allows a spontaneous decision process; it does rely on the group of participants only and does not require an election commission (i.e., a central authority that handles the election and determines the results) and it realises the security requirements described in section 2.

The “evote” protocol (described in section 4) fulfils all these requirements: All members have the same rights to start the decision process, to check if their votes are counted and to determine the result. Furthermore a spontaneous decision is supported, because no PKI (with the corresponding setup procedures) is required. Instead, the encryption and signing of votes is implemented with the keys that result from the DH-Key Agreement.

5 Conclusions

In this paper we discussed the use of Instant Messaging Systems as a platform for collaborative applications with high security requirements. As an example for such application, we defined scenarios of decision processes, which include a discussion and a subsequent e-voting. We presented a simple architecture and a corresponding implementation of our approach and discussed the first phase of the whole process, the authentication mechanism, which is based on a combination of DH key agreement and personal recognition of participants in a video stream. Finally, we presented a simple group-oriented voting protocol which is especially suitable to be included in our scenarios of discussion and decision in small groups via IMS. Our main finding is that this combination provides a *secure* and *spontaneous* way for binding collaboration in small groups such as decision committees in universities or other organisations.

Reference

1. Vaishnavi, V., Kuechler, W.: Design Research in Information Systems (2004/2005), <http://home.aisnet.org/displaycommon.cfm?an=1&subarticlenbr=279> (2008-09-09)
2. Universität Koblenz-Landau: Grundordnung der Universität Koblenz-Landau, Universität Koblenz-Landau (2006), https://www.uni-koblenz.de/gesetze/dateien/uni/uni_grundo_20060323.pdf (2009-01-17)
3. Universität Koblenz-Landau: Wahlordnung für die Wahlen der Organe der Universität Koblenz-Landau, 21 Staatsanzeiger Staatsanzeiger (Hrsg.) (2006), <http://zopeone.uni-koblenz.de:8080/CMS/unikold/verwaltung/abt-3/ref32-recht/wahlo/view> (2009-01-17)
4. Skype: Skype - take a deep breath. Skype Technologies S.A., <http://www.skype.com> (2009-01-17)
5. Spark: Spark - a jive software community, ignite realtime, <http://www.igniterealtime.org/> (2008-03-28)
6. Openfire: Openfire - a jive software community. Ignite Realtime Community, <http://www.igniterealtime.org/> (2009-01-17)
7. Diffie, W., van Oorschot, P.C., Wiener, M.J.: Authentication and Authenticated Key Exchanges. In: Designs, Codes and Cryptography, vol. 2, pp. 107–125. Kluwer Academic Publishers, Dordrecht (1992)

8. Zimmermann, P.: ZRTP: Media Path Key Agreement for Secure RTP, <http://www.zfoneproject.com/docs/ietf/draft-zimmermann-avt-zrtp-04.pdf> (2009-01-17)
9. Benaloh, J.C., Yung, M.: Distributing the power of a government to enhance the privacy of voters. In: Proceedings of the fifth annual ACM symposium on principles of distributed computing, Calgary, Alberta, Canada, pp. 52–62. ACM, New York (1986)
10. DeMillo, R.A., Lynch, N.A., Merritt, M.I.: Cryptographic Protocols. In: Proceedings of the 14th ACM Symposium on the Theory of Computing, pp. 383–400. ACM, New York (1982)
11. Alkassar, A., Krimmer, R., Volkamer, M.: Online-Wahlen für Gremien. In: DuD Datenschutz und Datensicherheit, vol. 8(29) (2005), <http://www2.dfki.de/fuse/dud2005.pdf> (2009-01-17)

A Multi-criteria Collaborative Filtering Approach for Research Paper Recommendation in Papyres

Amine Naak, Hicham Hage, and Esma Aïmeur

Université de Montréal PO BOX 6128, Station: Centre-Ville
Montréal Qc, H3C 3J7 Canada
{naakamin,hagehich,aimeur}@iro.umontreal.ca

Abstract. Graduate students, professors and researchers regularly access, review, and use large amounts of literature. In previous work, we presented Papyres, a Research Paper Management Systems, which combines bibliography functionalities along with paper recommender techniques and document management tools, in order to provide a set of functionalities to locate research papers, handle and maintain the bibliographies, and to manage and share knowledge about the research literature. In this work we detail *Papyres*' paper recommendation technique. Specifically, Papyres employs a Hybrid recommender system that combines both Content-based and Collaborative filtering to help researchers locate research material. Particularly, in this work special attention is given to the Collaborative filtering process, where a *multi-criteria* approach is used to evaluate the articles, allowing researchers to denote their interest in specific parts of articles. Moreover, we propose, test and compare several approaches to determine the neighbourhood in the Collaborative filtering process such as to increase the accuracy of the recommendation.

Keywords: Multi-criteria recommendation, collaborative filtering, research paper recommendation, eEducation.

1 Introduction

In any research field, graduate students and professors (henceforth referred to as researchers) regularly access, read and keep research papers of interest. With the advent of digital libraries, such as the *ACM Digital Library*¹, *IEEEExplore*² and *SpringerLink*³, most of these research papers can be accessed online and stored on personal computers in an electronic format. The storage and manipulation of these articles in their electronic format is namely easier than in their traditional paper format. As such, researchers regularly perform the following actions on research literature: locate new literature, manage the references, and manage the documents and their knowledge about these documents. Yet, to the best of our knowledge, there are no standalone solutions that

¹ <http://portal.acm.org/dl.cfm>

² <http://ieeexplore.ieee.org>

³ <http://springerlink.metapress.com>

answer to all the researchers' needs. Specifically, bibliography management systems, such as *EndNote*⁴, *CiteULike*⁵ and *BibTeX Tools*⁶ (*BibTool*, *Bibshare*, *WinBib*, etc.), help manage the references, and seamlessly format article citations into various formats (IEEE, APA, etc.). Nonetheless, bibliography management systems do not provide functionalities to locate and manage new literature. On the other hand, recommender systems, such as *Knowledge Sea II* [1] and *TechLens* [2, 3], offer research paper recommendations based on various considerations, but do not provide tools to manage the references and the literature. Alternatively, ECM (Enterprise Content Management) systems, such as *LiveLink* [4], offer tools and functionalities to manage document and users' implicit knowledge. Nonetheless, these tools are designed for a corporate environment, and are created specifically to manage internal documents, projects, workflow, etc. Thus, such systems' functionalities are not directly related to research, and do not take into account the specificities of research papers.

As such, in [5] we introduce *Research Paper Management Systems*, which combine the bibliography functionalities along with paper recommender techniques and ECM (Enterprise Content Management) document management tools, in order to provide a set of functionalities to locate research papers, handle and maintain the bibliographies, and to manage and share knowledge about the research literature. Specifically, our system *Papyres* is intended to illustrate Research Paper Management Systems. Moreover, *Papyres* promotes the Web2.0 *harnessing the collective intelligence* approach [6], and employs Web2.0 techniques and technologies such as *Tagging*, *Rating*, and *RSS* (Really Simple Syndication). In this work, we elaborate further the paper recommendation system of *Papyres*. Specifically, we highlight the specificities of recommending research literature, as well as why the existing paper recommendation techniques do not suit our needs. In particular, when searching for articles, there are two factors to consider: the article's *content* and its *quality*. The content of the article is not limited to its actual content and topic of the article, but extends to other features including the author, the date, the conference/journal, etc. On the other hand the *quality* of an article is not necessarily an overall score or evaluation of the article, but can be relative to the objectives of the search and the needs of the researcher. For instance, an article with a modest *contribution*, but with detailed information on the *implementation* might be poorly appreciated by a researcher who is interested in innovative new research. Yet, the same article might be highly appreciated by another researcher interested in the practical aspects of the theory, specifically the implementation. With this in mind, the paper recommendation technique used within *Papyres* combines *Content-based* filtering (CBF) along with *Collaborative filtering* (CF) techniques in order to perform the recommendation. Specifically, the CF technique, which is the main focus of this article, utilizes a *multi-criteria* recommendation approach [7, 8] to provide the flexibility required in order to evaluate the quality of articles depending on the objective of the search. Additionally, in order to increase the accuracy of the recommendation, we propose and explore various methods to determine the neighborhood on which CF recommendations are based.

⁴ <http://www.endnote.com/>

⁵ <http://www.citeulike.org/>

⁶ <http://www.bibtex.org/>

The paper is organized as follows: section 2 highlights the various considerations when searching for research literature. Section 3 offers some background information on Papyres, and offers an overview of existing article recommendation systems. Section 4 details our approach and section 5 highlights the testing procedure and results. Section 6 concludes the paper and provides an overview of future work.

2 Locating Research Literature

When searching for research literature or articles, one must consider two aspects: the content of the article, and its quality. Specifically, when searching for literature, researchers usually have a particular idea of their need, and a certain amount of information about the content of the intended goal. For instance, a researcher might be looking for articles on intelligent agents, or more specifically intelligent agents used specifically in ITS (Intelligent Tutoring Systems). Alternatively, the researcher might have a more explicit goal in mind: articles on ITS published in 2007, at a certain conference, or all the journal articles published by the specific author “John Doe” on intelligent agents. As such, the content of the research literature does not relate exclusively to the actual subject matter, or topic of the article, but extends to other aspects as well including the author(s), the type of article (conference, journal, etc.), the date of publication, etc.

On the other hand, *quality* is also an important factor to consider when searching for literature. Indeed, although a research paper might satisfy the *content* requirements, its *quality* might not satisfy the researcher’s expectations, nor answer his needs. Specifically, the *quality* of a research paper is relative and does not necessarily relate to an “overall quality”. Indeed, in a recent survey we conducted [5], 84% of the respondents (composed of graduate students and professors) reported that, when searching for literature, they are regularly interested in a specific part of the article. As such the *quality* of that *specific part of interest* is the most important and not the overall quality. For instance, consider a researcher who is interested in doing some background research on a specific subject. Consequently, research papers with a good state of art will satisfy the researcher’s *quality* requirement, whether or not the proposed work in these articles is of any significance. Hence, the researcher should be able to locate new literature by specifying his *quality requirements* for one or several parts of an article.

3 Background

In this section we start by briefly introducing Papyres, then we offer an overview of existing paper recommender systems and highlight why these systems do not meet the expectations defined in the previous section.

3.1 Papyres

Papyres aims to help researchers manage and make the most of their research resources (research papers, reports, e-books, etc.). Specifically, Papyres combines functionalities from bibliography management systems with paper recommendation

techniques and ECM document management tools along with Web2.0 techniques to offer researchers a complete environment for managing research literature as well as harnessing and sharing knowledge. **Fig. 1** offers a highlight of Papyres process. The first step in using Papyres, of course other than registering in the system, is to add research papers, or resources. When adding a new resource, researchers must specify its *availability*. Indeed, researchers have the option to keep a resource *private*, make the resource *available* to one or more *groups*, or *open* to any user within Papyres. Moreover researchers can add *notes* and *comments*, attach these comments to specific sections of the article, and even share these notes and comments with others.

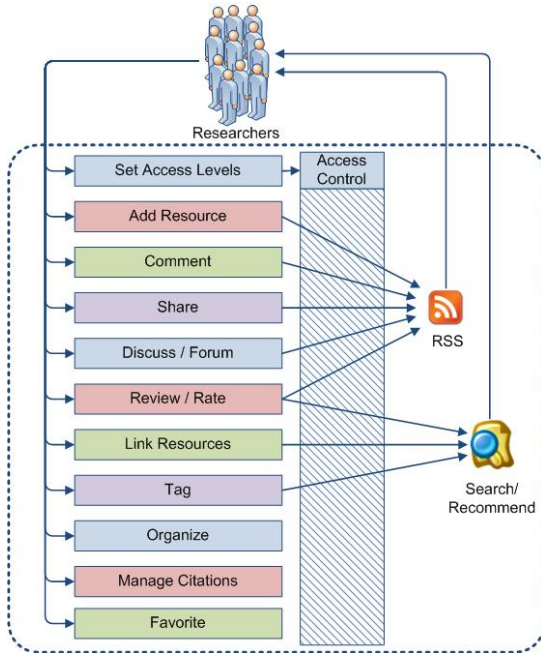


Fig. 1. Papyres process

In addition, researchers can review and evaluate articles in two complementary methods. First, researchers can write a general review text, or they can attach various review texts to different sections of the resource. For instance, the researcher can separate the review into parts *dedicated* to specific sections of the resource. As such, the researcher can write a review text for the State of Art, and another review text for the Proposed Approach. The purpose of such a partition is to provide a better flexibility and to make it easier to evaluate a certain resource. Indeed, if a researcher is interested in a certain part of a resource, checking the reviews related to that specific part would be informative enough.

Second, the researcher is presented a set of evaluation questions pertaining to various parts of the resource. A total of 10 evaluation questions rate, on a scale of 1 to 5, various aspects of articles including: *Contribution*, such as the significance of the problem addressed in the resource, and the significance of the proposed solution to

the research community; *Originality*, to evaluate the novelty of the proposed work; *Literature Review*, to assess if the literature review offers a clear and proper background, and how comprehensive it is; *Readability* and *Organization*, to evaluate if the resource is clearly written and its sections well structured; *Technical Quality*, to assess if the proposed approach is clearly explained and detailed, and if the approach is technically sound; *Testing, Procedure*, to evaluate if the testing procedure is well defined, and if the reported result properly justify the hypotheses; *Quality of References*, to assess if the references are up to date, or if any are missing.

Additionally, Papyres provides a discussion forum where a group of researchers can discuss and evaluate articles, as well as the possibility to *tag* articles or *link* them with one of the following relationships: *support*, *complement*, *criticize*, *contradict*, *illustrate*, *implement* or *similar*. Moreover, Papyres allows researchers to *organize* their articles into categories or folders, to *manage the bibliography*, as well as the *favorite* articles.

On the other hand, Papyres provides a complex set of RSS (Really Simple Syndication) feeds to update researchers with various information on resources. RSS is a family of Web feed formats used to publish frequently updated content. An RSS document, which is called a feed, usually contains either a summary of content, from the associated web site or the full text. In short, RSS makes it easier to keep up with updates. Particularly, Papyres uses RSS feeds to advise researchers with updates of their interest. Moreover, Papyres offers a recommendation system, detailed further in section 0, to help researchers locate new articles.

3.2 Paper Recommender Systems

With the proliferation of research conferences and journals, and wide range of available research papers, it becomes harder for researchers to easily locate resources that suites their needs. Indeed, the number of scientific articles published in internationally recognized peer-reviewed Scientific & Engineering journals covered by the Science Citation Index (SCI) and Social Sciences Citation Index (SSCI) was nearly 700,000 in 2003 [9]. Hence the various attempts at using recommender techniques [1, 3, 10] to help researchers locate suitable research material. For instance, Knowledge Sea II [1] treats research papers as regular pedagogical resources, allowing users to annotate and review these resources, and using the annotations to perform the recommendations. As such, Knowledge Sea II relies on the content of articles (reflected in the tags) to perform the recommendations. TechLens [3] on the other hand combines the citations of an article along with its content in order to perform the recommendation. Moreover, Kapoor *et al.* [2] propose to enhance TechLens by using the user's personal citation libraries to enhance his profile and produce the recommendation. Again in this case the recommendation is mainly based on the content of the article. Recently, Matsatsinis *et al.*[9] and Lakiotaki *et al* [8] proposed a new approach: Multiple-Criteria Decision Aiding (MCDA). This approach is based on decision making theory, a widely used approach in the Operations Research field. Alternatively, Tang *et al.* [11, 12] propose a recommender system which takes into account pedagogical aspects. In short, the recommender system considers the knowledge level of the students in the recommendation process. Comtella [10] is another academic system that uses P2P (peer to peer) technology to enable students to share research papers. In addition, Comtella employs a reputation scheme [13] to motivate and award the students.

On the other hand, systems such as CiteSeer [14] and Google Scholar⁷ are search engines dedicated for research papers. Specifically, CiteSeer concentrates primarily on literature in computer and information science, providing citation analysis such as how many times a paper is referenced, and in which other articles. Alternatively, Google Scholar provides a wider range of research articles, and sorts the articles by weighing the full text of each article, the author, the publication in which the article appears, and how often the article has been cited in other scholarly literature.

In summary, the existing recommender systems either are intended to recommend articles to learners, or mainly rely on the content of the articles, and in some cases on citations to create further links between articles' contents. Moreover, when ratings are used to evaluate the quality of an article, the exiting solutions rely on an overall rating of the article, which is needless to say ineffective in our case for the following reasons. First, an overall rating of an article does not provide the required granularity and flexibility such as to allow the evaluation of the quality of specific parts of interest within the article. Second, an overall rating of an article can produce a bias in the recommendation, specifically in the case of collaborative filtering. Indeed, two researchers might give the same evaluation score for the same article, but for completely different considerations.

4 Multi-criteria Hybrid Recommender System

In this section we introduce the recommendation technique used within Papyres. The approach is tailored to answer the researchers' requirements highlighted in Section 0. Papyres recommender system is a cascade Hybrid [15], where the first level is a Content-based filter (CBF), and the second level uses Collaborative filtering (CF) techniques.

4.1 Content Based

When looking for articles, researchers usually have some knowledge of the desired goal. This information about the intended goal can be as vague as a general idea such as any article on recommender systems, or very specific such as articles on recommender systems by the author John Doe published during a certain period (for example between 2002 and 2007). Papyres provides such flexibility as to allow researchers to specify one or several content criteria based on the various meta-data stored on the articles. Specifically, Papyres stores various meta-data inspired by LOM (Learning Object Metadata) [16], on research literature. Fig. 2 highlights the specific criteria used within the CF grouped into two categories: the General and the Bibliography categories. In particular, the general category contains general information about the article, and the researcher can specify one or more *search words* to be matched in either the *title*, the author specified *keywords*, or the users specified *tags*. It is important to note the difference between the keywords and the tags. The keywords are usually specified by the author of an article when it is submitted for publication, whereas

⁷ <http://scholar.google.ca/>



Fig. 2. Content search criteria

the tags are *attached* by the users of Papyres to the article. Additionally, the researcher can specify the *Language* of the desired article, as well as its *Type*. In this case, the type of the article reflects whether the article is a survey on a certain topic, a report on an experiment, if the article presents some new research, etc.

The bibliography category contains search information relating to the bibliographic data of an article. As such, the researcher can specify one or more authors, specific conference(s) or journal(s), a particular, or boundary year (articles published after the year 2000). Additionally, researchers can specify the type of the publication they are searching for, such as they can, for instance, specify that they are interested only in journal articles. In such a case, the CBF recommender processes the researcher's request, and retrieves all the articles that meet the search criteria. Nonetheless, the CBF recommendation returns large sets of possible articles, specifically when only few search criteria are specified by researchers. Consequently we combine another technique to further enhance the efficiency of the recommendation. Papyres takes the set of articles retrieved by the CBF and uses a CF approach to *estimate* the *quality* of the articles.

4.2 Multi-criteria Collaborative Filtering

A Collaborative Filtering recommender system accumulates user ratings of items, identifies users with common ratings, and offers recommendations based on inter-user comparison. As such, the CF recommender compares researchers' ratings of articles, and tries to determine the rating the researcher will give to the current article, determining whether the resource will satisfy the researcher or not. The CF process is composed of two steps: first determine the neighborhood of the researcher, which consists of k researchers with the highest ratings' similarity. The similarity between the researcher a and his neighbor u is derived using Pearson correlation coefficient highlighted in the following equation:

$$sim_{a,u} = \frac{\sum_{i=1}^m (r_{a,i} - \bar{r}_a) \times (r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i=1}^m (r_{a,i} - \bar{r}_a)^2 \times \sum_{i=1}^m (r_{u,i} - \bar{r}_u)^2}} \quad (1)$$

Where $r_{a,i}$ is the rating given by researcher a to article i , r_a is the mean rating given by researcher a , and m is the total number of articles. The next step is to use this neighborhood to *predict* the researcher’s rating to an unrated article. Such predictions are computed as the weighted average of deviations from the neighbor’s mean using the following equation:

$$p_{ai} = \bar{r}_a + \frac{\sum_{u=1}^k (r_{u,i} - \bar{r}_u) \times sim_{a,u}}{\sum_{u=1}^k sim_{a,u}} \tag{2}$$

Where p_{ai} is the rating prediction of researcher a for resource i , $sim_{a,u}$ is the similarity between researchers a and u obtained using equation (1), and k is the number of researchers in the neighborhood. As recommended in [17], we use a neighborhood of $k = 30$ researchers. Fig. 3 [18] illustrates the CF recommendation that is usually used and based on a single rating per item setting.

	Item i_1	Item i_2	Item i_3	Item i_4	Item i_5	
Target user User u_1	5	7	5	7	?	
Users most similar to the target user	User u_2	5	7	5	7	9
	User u_3	5	7	5	7	9
	User u_4	6	6	6	6	5
	User u_5	6	6	6	6	5

Fig. 3. CF in a single rating setting [18]

Such an approach is inefficient in the context of article recommendation for two reasons. First, a single rating per article does not provide the required granularity and flexibility such as to allow the evaluation of the quality of specific parts of interest within the article. Second, an overall rating of an article can produce a bias in the recommendation: two researchers might give the same evaluation score for the same article, but for completely different considerations. As such, we use a multi-criteria approach inspired by [7], where researchers provide several ratings for each article, each evaluating a certain aspect of the article, including the *Contribution* (the significance of the problem addressed in the resource, and the significance of the proposed solution to the research community), *Originality* (to evaluate the novelty of the proposed work), *Literature Review* (to assess if the literature review offers a clear and proper background, and how comprehensive it is) *Readability* and *Organization* (to evaluate if the resource is clearly written and its sections well structured), *Technical Quality* (to assess if the proposed approach is clearly explained and detailed, and if it is technically sound), *Testing Procedure* (to evaluate if the testing procedure is well defined, and if the reported results properly justify the hypotheses), and *Quality of References* (to assess if the references are up to date, or if any are missing). In this case, by comparing how two researchers rate the various aspects of an article, the CF

recommender can determine more accurately how similar the two researchers are. Additionally, this granularity provides the possibility to evaluate the *quality* of the article with respect to the researcher's objectives and needs. In contrast to an overall rating, the granularity of the ratings enables, for instance, a researcher who is an expert in a domain to search for articles with a high *Contribution* and *Originality*, and for new researchers to search for articles with a good *Literature Review* and a high *Quality of References*. Fig. 4 [18] illustrates the multi-criteria approach in contrast to the single criteria. In this case, the overall rating of an item (the larger numbers) are broken down to four ratings on each item. Notice that although the overall ratings (large numbers) of the users u_2 and u_3 are closer to the overall ratings of the target user u_1 , they have rated the various aspects (smaller numbers) of the item in a completely opposite manner to u_1 , thus u_2 and u_3 actually have opposite preferences to u_1 . Consequently, u_4 and u_5 have *closer* preferences when considering the multi-criteria ratings, and are more suited to perform the recommendation.

Within CF, it is imperative to properly determine the neighborhood, since the accuracy of the rating's prediction relies heavily on this neighborhood, and how *close* the neighbors are to the target user. Hence we propose and compare five various approaches at determining the neighborhood.

The **first approach**, which we refer to as the horizontal (**HZ**) approach, is primarily based on the work presented in [18]. The similarity between the target researcher a and the potential neighbor i is determined for each of the 10 rating criteria, and then the average of these *partial* similarities is used to determine a *global* similarity. The k users with the highest global similarity are then used in the prediction process. Although this approach tends to minimize the overall inaccuracy, it does introduce some *noise* in the data, specifically when a neighbor with a high global similarity is very divergent on one or more criteria. For instance researcher a and his neighbor u rated most criteria in a similar manner except for 2. When predicting the rating of a , the divergence of u will introduce some noise/inaccuracy in the process, even though the similarity is used as a weighing factor (refer back to equation (2)).

	Item i_1	Item i_2	Item i_3	Item i_4	Item i_5
Target user User u_1	5 _{2,2,8,8}	7 _{5,5,9,9}	5 _{2,2,8,8}	7 _{5,5,9,9}	?
Users most similar to the target user User u_2	5 _{8,8,2,2}	7 _{9,9,5,5}	5 _{8,8,2,2}	7 _{9,9,5,5}	9
User u_3	5 _{8,8,2,2}	7 _{9,9,5,5}	5 _{8,8,2,2}	7 _{9,9,5,5}	9
User u_4	6 _{3,3,9,9}	6 _{4,4,8,8}	6 _{3,3,9,9}	6 _{4,4,8,8}	5
User u_5	6 _{3,3,9,9}	6 _{4,4,8,8}	6 _{3,3,9,9}	6 _{4,4,8,8}	5

Fig. 4. CF in a multi-criteria rating setting [18]

In the **second approach**, that we refer to as the vertical (**VL**) approach, the predictions for each of the criteria is performed in the same manner as in the *classical* single criteria rating settings. In other words, for each criteria j , the k users with the highest similarity in rating the criteria j are used in the prediction. The rationale behind this approach is to use the closest neighbors for each criteria instead the closest neighbors overall. As such, this approach takes advantage of the multi-criteria setting, while maximizing neighborhood's similarity with the target researcher for each criterion separately. Yet this approach suffers greatly when the neighborhood is not very close to the target user.

The size of the neighborhood as well as its similarity to the target researcher is important. When using Pearson correlation coefficient, a similarity value between 0.5 and 1 implies a high correlation, and a similarity value between 0.3 and 0.5 implies a medium correlation. As such, we set a similarity threshold T to 0.3, such that the neighborhood of a target researcher is close. Nonetheless, applying the threshold alone on the previous approaches is not sufficient, since the size of the neighborhood affects the accuracy of the prediction. Indeed, setting the threshold for the HZ and VL approaches reduced the neighborhood as well as the average performance. As such, in order to complement the neighborhood while maintaining a high similarity, we propose the next two approaches.

In the **third approach**, that we refer to as the horizontal then vertical (**HZ-VL**) approach, the neighborhood is composed first by the most similar neighbors whose average similarity is larger than T . If the number of neighbors is less than k , then the neighborhood for each criteria j is complemented by the neighbors with highest similarity to a with regards to criteria j and always larger than T .

In the **fourth approach**, that we refer to as the vertical then horizontal (**VL-HZ**) approach, the neighborhood is determined in a similar manner as the previous approach, but in this case it is determined *vertically* first, then complemented *horizontally*.

The **fifth approach**, which we refer to as horizontal without noise (**HZ-N**), is actually an enhancement of the first approach, since we expect to have some *noise* in the first approach. Specifically, consider a target user a and one of his k nearest neighbors i . When performing the prediction for criteria j , the user i will be considered in the prediction, even though his similarity with a for that specific criteria j is not close, hence introducing some *noise*. We consider such cases as *noisy* data that we simply disregard from the computation. We determine noisy data again by comparing the similarity to the threshold $T = 0.3$.

5 Testing and Result

In order to test and compare the accuracy of the five proposed techniques, we use a leave one out approach. Specifically, we randomly select a researcher and an article rated by that researcher. Afterwards we assume that the researcher hasn't rated the article yet, and we attempt to predict his ratings. Finally we compare the *predicted* with the *actual* ratings in order to evaluate the accuracy of the prediction. The MAE (Mean Absolute Error) is a metric regularly used in order to evaluate the accuracy of such predictions [7, 19]. The MAE is derived using the following equation:

$$MAE = \frac{1}{n} \sum_{i=1}^n |f_i - y_i| \quad (3)$$

Where n is the number of predictions, f_i is prediction i and y_i is the actual value. In short, MAE presents the *average* difference between the prediction and the actual value.

The dataset on which the tests were performed was built artificially in a pseudo-random manner. Specifically, Papyres is still in its development phase and does not contain sufficient data in order to perform valid tests and draw conclusive results. Alternatively, we contacted Yahoo! (during November 2008) to obtain their movie ratings dataset. Specifically, users can rate movies on the Yahoo! website with regards the following four criteria: Story, Acting, Direction and Visual, as well as provide a fifth, overall rating. However, the currently available dataset contains only the overall rating per movie. Specifically here is part of the response we received from Yahoo!: “I just chatted with the research scientist and as I was afraid, we don't have the data in the configuration that you'd like”.

The dataset was built in two steps. First, a set of 20 users were created. Then, for each user, ratings for 30 different articles were specified randomly. In order to reduce the random effect of the ratings and to create a certain correlation between the users, we augment the initial dataset of 20 users in a pseudo-random manner. As such, for each of the *initial users*, we create 10 *additional users* whose ratings are based on the ratings of the initial user, varied in a *logical*, but *random* manner. For instance, consider that $R_{a,i}$ is the set of ratings of initial user a for article i . Then the set of ratings of new user j is $R_{a,i} + \text{Random values from } \{-1,0,1\}$. That is the ratings of j are equal to the ratings of $a \pm 1$. In summary, based on each of the initial 20 users, 10 more users were created where their ratings *profile* was randomly chosen from the following sets: $\{-1,0,1\}$, $\{-2,-1,0\}$, $\{0,1,2\}$, $\{-3,-2,-1\}$, $\{1,2,3\}$, $\{-1,0,1,2\}$, $\{-2,-1,0,1\}$, where each set is at least chosen once in each case. As such, the dataset was composed of 220 different users (20 initial users + 200 additional users) where each rated 30 articles.

5.1 Results and Findings

In order to test the approaches, a test set of 100 different researcher/paper pairs were selected randomly. Afterwards, the five approaches were utilized to predict the ratings of the test set, and then these predictions were compared to the actual ratings using MAE. The MAE of each criterion is recorded, as well as the average MAE over all the criteria. The average MAE over the 100 iterations is used to compare the performance of the various implemented approaches. **Fig. 5** highlights the best case, the worst case and the average case over the 100 iterations of the five approaches: the minimum MAE (MIN MAE), maximum MAE (MAX MAE) and the average MAE (AVG MAE) respectively. Overall, the least performing approach is the VL approach. Specifically, this approach suffered mainly in cases where the similarity between the researcher and his neighborhood is not *close* enough. The HZ approach addresses this issue where the global similarity is considered. As such, even when the researcher's neighborhood is somewhat far for a specific criterion, the overall similarity reduced the overall error in the predictions. On the other hand both HZ-VL and VL-HZ

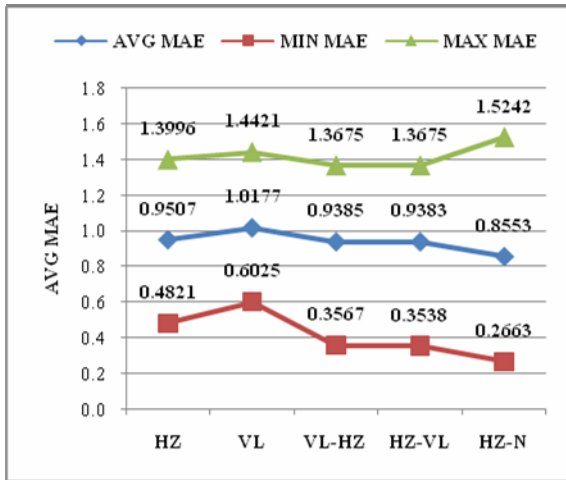


Fig. 5. MAE comparison

maximize the similarity of the neighborhood, and do offer an overall better performance over HZ and VL separately. On the other hand, although HZ-N has the highest MAX MAE, this approach still offers the best overall performance. HZ-N takes advantage of the overall similarity, while reducing the *noise* induced by neighbors with a high overall similarity, but who are not very similar for a certain criterion.

In order to interpret the values of MAE, it is important to consider the scale on which the ratings are performed. Indeed, an MAE of 0.5 indicates that the predictions, on average, differed by 0.5 of the actual rating. In order to evaluate the impact of this difference, it is important to consider the scale of the predictions. Indeed, a difference on 0.5 on a scale of 1 to 5 is more significant than on a scale of 1 to 20. As such we compare the variation, or the MAE, to the scale to assess its actual impact; that is a MAE of 0.5 on a scale of 5 represents 10% whereas on a scale of 20, it only represents 2.5% and consequently a lower impact on accuracy. Fig. 6 highlights the average MAE of the five approaches evaluated on a scale of 5.

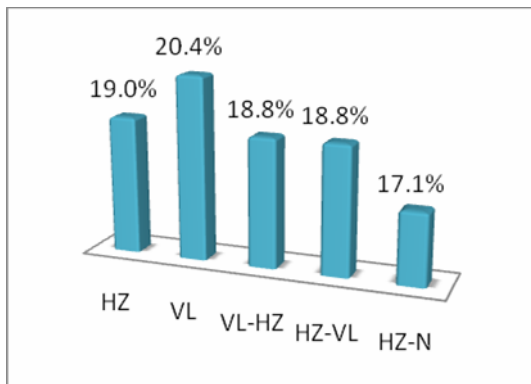


Fig. 6. Average MAE interpretation

Although the results are encouraging, and the HZ-N does offer further enhancement of the accuracy over the other approaches, an MAE of 0.8 (or a variation of 17%) leaves some space for improvements. Nonetheless, we presume that the MAE of 0.8 is due essentially to the fact that the dataset is made of randomly generated data. Indeed, we believe that the value of the MAE will be smaller when testing the approaches on a real dataset. Furthermore, we also believe that the HZ-N will perform better than the other approaches, since all the testing was executed on the same dataset.

6 Conclusions and Future Work

Recommending research literature involves two aspects: the content and the quality of an article. The content extends to more than the actual subject matter, and covers other characteristics including the author(s) or the year of publication. The quality on the other hand does not refer to an overall quality, but rather an *objective* quality depending on the needs to the researcher. Existing solutions do not properly address the *quality* issue, the main focus of this article. Indeed, in this work we detail a multi-criteria Collaborative-filtering (CF) approach to satisfy the researchers' quality requirement. Specifically, we introduce new approaches to determine the neighborhood for the Collaborative Filtering recommender system. Although we describe the approaches in the context of recommending research literature, it is important to note that these proposed approaches can be applied to any case of multi-criteria collaborative filtering. In addition, using a pseudo-randomly generated dataset, we compare the three proposed approaches (**HZ-VL**, **VL-HZ** and **HZ-N**), to the *classical* CF approach (**VL**), as well as the *plain multi-criteria* approach (**HZ**). Preliminary results and findings are encouraging where the HZ-N offers the best performance.

We do agree with Herlocker [19] that generated datasets are to be used for *preliminary* results, and that further testing should always be performed for more *conclusive* results and conclusions. Nonetheless, we do corroborate the validity of our preliminary results. Indeed, comparing the performance of several approaches on the same dataset, even though it is a pseudo-randomly generated dataset, does offer some validity to the results. Moreover, the random factor provides *neutrality* to the dataset, such as it is not designed specifically to enhance the performance of one approach while deteriorating the performance of another. In addition, the pseudo-random aspect of the dataset ensures a proper distribution of profiles and similarities over a wide range. In particular, the similarity based on Pearson's correlation (equation (1)) varies between -1 (completely different) and 1 (perfect similarity), while the similarity between the pseudo-randomly generated users varies between -0.804 and 0.941 covering most of the spectrum of values.

Another part of the recommender system that was not detailed in this work is the content-based filter (CBF). Specifically we are investigating the adaptation and/or combination of some of the existing techniques (highlighted in section 0) into the

CBF in order to increase the efficiency and accuracy of the hybrid recommender system. However, we leave this for future works.

References

1. Brusilovsky, P., Farzan, R., Jae-wook, A.: Comprehensive personalized information access in an educational digital library. In: 5th ACM/IEEE-CS Joint Conference on Digital Libraries, Denver, CO, USA, pp. 9–18 (2005)
2. Kapoor, N., Chen, J., Butler, J.T., Fouty, G.C., Stemper, J.A., Riedl, J., Konstan, J.A.: Techlens: a researcher's desktop. In: ACM Conference on Recommender Systems (RecSys 2007), Minneapolis, MN, USA, pp. 183–184 (2007)
3. Torres, R., McNee, S., Abel, M., Konstan, J., Riedl, J.: Enhancing digital libraries with TechLens+. In: 4th ACM/IEEE joint conference on Digital libraries (JCDL 2004), Tuscon, USA, pp. 228–236 (2004)
4. Huang, X., An, A., Cercone, N., Promhouse, G.: Discovery of Interesting Association Rules from Livelink Web Log Data. In: IEEE International Conference on Data Mining (ICDM 2002), Maebashi, Japan, pp. 763–766 (2002)
5. Naak, A., Hage, H., Aïmeur, E.: Papyrus: a Research Paper Management System. In: IEEE Joint Conference on E-Commerce Technology (CEC 2008) and Enterprise Computing, E-Commerce and E-Services (EEE 2008), Crystal City, Washington (2008)
6. O'Reilly, T.: What Is Web 2.0 (2005) (accessed, January 2008), <http://www.oreillynet.com/>
7. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17(6), 734–749 (2005)
8. Lakiotaki, K., Tsafarakis, S., Matsatsinis, N.: UTA-Rec: a recommender system based on multiple criteria analysis. In: ACM Conference on Recommender Systems (RecSys 2008), Lausanne, Switzerland, pp. 219–226 (2008)
9. Matsatsinis, N.F., Kleanthi, L., Pavlos, D.: A System based on Multiple Criteria Analysis for Scientific Paper Recommendation. In: 11th Panhellenic Conference in Informatics, Patras, Greece, pp. 135–149 (2007)
10. Vassileva, J.: Harnessing P2P Power in the Classroom. In: Lester, J.C., Vicari, R.M., Paragauçu, F. (eds.) ITS 2004. LNCS, vol. 3220, pp. 305–314. Springer, Heidelberg (2004)
11. Tang, T.Y.: The design and study of pedagogical paper recommendation. Doctoral Thesis. University of Saskatchewan, Saskatoon (2008)
12. Tang, T.Y., McCalla, G.I.: A multi-dimensional paper recommender. In: 13th International Conference on Artificial Intelligence in Education Marina Del Rey, Los Angeles, California, U.S.A (2007)
13. Mao, Y., Vassileva, J., Grassmann, W.: A System Dynamics Approach to Study Virtual Communities. In: 40th Annual Hawaii International Conference on System Sciences, Wai-koloa, Hawaii, p. 178a (2007)
14. Bollacker, K., Lawrence, S., Giles, L.: A System for Automatic Personalized Tracking of Scientific Literature on the Web. In: 4th ACM Conference on Digital Libraries, Berkeley, California, United States, pp. 105–113 (1999)

15. Burke, R.: Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction* 12(4), 331–370 (2002)
16. IEEE Standard for Learning Object Metadata. IEEE Std 1484.12.1-2002, i-32 (2002)
17. Melville, P., Mooney, R.J., Nagarajan, R.: Content-boosted collaborative filtering for Improved recommendations. In: 18th National Conference on Artificial Intelligence, Edmonton, Alberta, Canada, pp. 187–192 (2002)
18. Adomavicius, G., Kwon, Y.: New Recommendation Techniques for Multicriteria Rating Systems. *IEEE Intelligent Systems* 22(3), 48–55 (2007)
19. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.* 22(1), 5–53 (2004)

An Ontological Approach to Connecting SOA Implementations

Wesley McGregor

CGI Group Inc

1 Introduction

1.1 Premise

Service Orientation as a design paradigm has taken root in the academic, development and business communities alike. The notion that activity models in all aspects of human endeavour can be thought of as a set of services with consumers and providers is not new, but the amount of “air time” the service oriented paradigm is getting is new. However like all good things in life, every good deed does not go unpunished. There are drawbacks to service oriented designs and this paper attempts to highlight one such drawback that all system practitioners must be aware of in the long term.

1.2 Purpose

The purpose of this paper is to illustrate a potential negative outcome and its mitigating solution to using Service Oriented Architecture (SOA) design principles and patterns without due consideration to the automation environment at large or the long term effects of isolated programming practices.

1.3 Scope

The scope of this paper is non-restrictive, that is, there is no boundary to which it cannot be applied. From massively interconnected systems to small in-home networks, all can benefit from the concepts presented herein.

1.4 Constraints

Due to time constraints, the author did not have sufficient time to explore nor provide adequate proof for most of the claims made in this paper. It is hoped that further research will be carried on by academics in the topics broached by this paper.

2 The Problem of SOA Discontinuity

2.1 The Theory

Any reasonable, useful and logical theory when first proposed naturally creates great interest and excitement. Service Oriented Architecture (SOA) is just such a theory. Numerous articles have been written about this “new” architecture and a significant

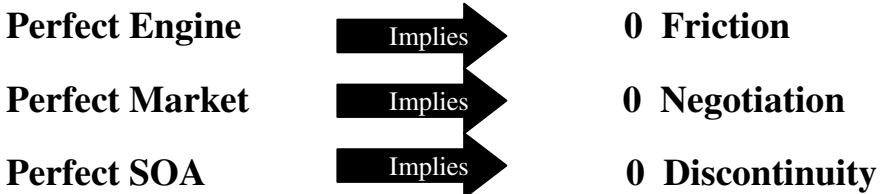
number of organizations are in the process of adopting it. However like any like new car or drug first introduced onto the market, sometimes there are a few latent issues which get discovered that can cause negative impacts to the consumer. SOA is no different and this paper discusses one such potential negative outcome if SOA designers and corporate architects do not keep a reign on their designs.

One of mankind's ever present ambitions is to build the perfect device. The well-known search for the perpetual motion machine is one such endeavour. In today's world, car manufacturers in conjunction with oil companies are constantly striving to create engines with close to zero friction. For those not familiar with car engines, the two largest causes of internal combustion engine inefficiency is the management of the heat created during the combustion process and the energy drained from the useful work component due to friction. To address these power losses, car manufacturers are constantly testing new lubricants designed by the oil companies to overcome and minimize the losses due to heat and friction.

In the marketplace, during the process of buying and selling of goods, a significant amount of energy is consumed (wasted if you will) in the process of negotiation. A consumer who wants to buy goods or services at a low price negotiates with the seller to minimize his costs while the seller attempts to maximize his profits without disenfranchising the potential buyer. The process of negotiation, although a necessary component of today's business, adds latency to the transaction, consumes energy in its execution and sometimes even results in a "no-deal" scenario.

As we all know, the advent of computer automation has significantly sped up the process of negotiation but nonetheless it still exists in a shortened form. Instantaneous buying and selling is not yet possible, but the reduction of the time it takes to perform these actions has been shortened to be close to zero. However a key thing to recognize is that not all types of commodities can have a zero negotiation time. For example, when an individual goes to a store to buy a hammer, the negotiation undertaken between the store and the consumer is zero, but when whole companies are purchased by other companies, the negotiation time can be quite large.

Service Oriented Architecture (SOA) also requires negotiation between the service provider and service consumer. Design time binding and even runtime binding takes time. Translation and transformation between different syntactical environments takes time. If we add to this the transformation of the meaning of things between environments more effort is required. This loss of time and effort can be collectively termed "discontinuity" and its cost can be quite large even within a single environment.



2.2 The Problem

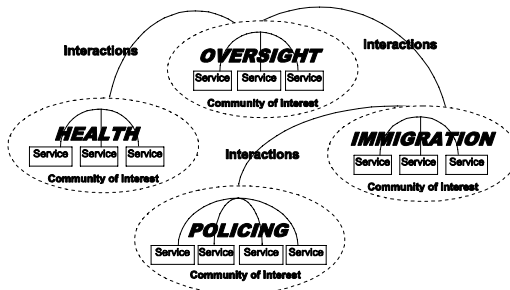
As stated, Service Oriented Architecture (SOA) can create inefficiencies and it is these inefficiencies which can ultimately lead to a problem known as discontinuity.

From a software construction standpoint, services are bound to their service descriptions and the service descriptions (such as those defined by the Web Services Description Language - WSDL) are then consumed by the Integrated Development Environments (IDEs) that understand them. This method of service consumption is pretty straight forward but for any service at an enterprise level typically a Service Level Agreement (SLA) is also required to guarantee (if that is possible) the performance characteristics and liabilities associated with that service. All told, what we have is an environment where large efforts are required to consume services. If the total effort required to create an SOA environment is labour intensive to say the least, how does SOA really help in assisting organizations bring their systems together especially if SOA promulgates isolationist systems?

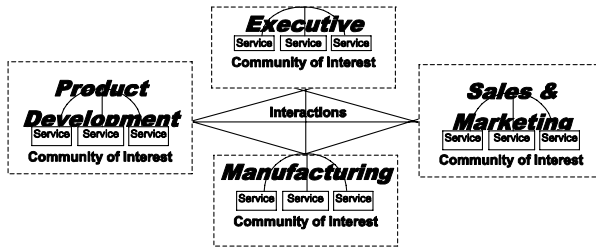
The Community of Interest model

The SOA environment has some inefficiency in design in order to provide a generalized pattern for interaction. This interaction is deemed syntactic interaction. There is also the often forgotten but equally or more important notion of the meaning of things – semantics. A service can only be negotiated for properly if the service in question is understood to be the same thing from both sides of the negotiation. For example, if I as a purchaser see a hammer as a carpentry device and the seller sees a hammer as a soil digging implement, the value and resultant utility of the hammer changes based on the perspective of the participant. And this contextualization of information only adds inefficiencies to the negotiations for the hammer should they even take place.

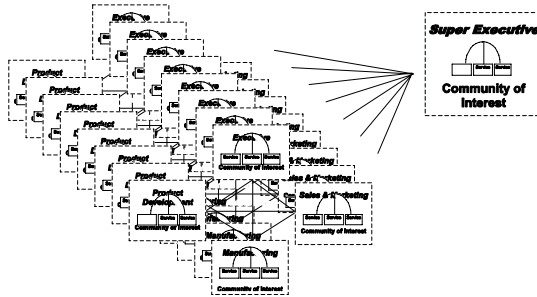
To overcome the issues of semantic incongruence, Communities of Interest (COI) usually are instantiated to provide a contextualized view where agreement on terms and their definitions is achieved well before any negotiations take place. For example, in the public service a subset of communities and their interactions look like the following:



In the private sector, in simple companies, Communities of Interest also spring forth. Small departments within a company typically become a community within the company and are usually required so that focus is maintained on key aspects of the overall business work flow.



If we extrapolate further and look at the giant conglomerate, we could have a plethora of communities, some that are very similar in nature, each residing in different companies probably duplicating efforts.



So how do communities within these organizational structures interact in an automated way within themselves and between themselves? The answer is that each community that is part of a conglomerate or each department within an organization uses local message streams, connection pathways, networks, remote method invocations or whatever is available in order to interconnect.

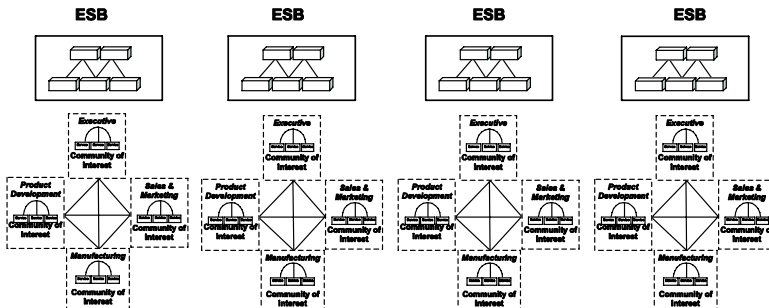
To summarize:

Using ESBs that provide...

- Registry/Repository
- Transformation
- Routing
- Reliable Messaging
- Standardized Interfaces
- Orchestration Engines
- Connectors & Adapters

enterprises end up with

- Localized syntax
- Localized nomenclature
- Localized semantics
- Vendor influences
- Interoperability challenges



In today’s marketplace, the set of messaging services complemented by a few other management services now come “out-of-the-box” and are offered as an Enterprise Service Bus (ESB). These ESBs are offered by a number of vendors and can be viewed as middleware for the service oriented world, but like any product offered by any vendor there are things to be aware of and things to consider.

3 Linking Communities - The Layered Ontological Overlay

3.1 General

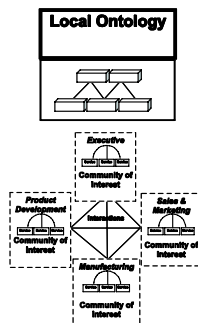
There are many definitions of the word ontology. Rather than inventing one here, a reference to Tom Gruber’s work for a definition will suffice for the purposes of our discussion.

Simply “An ontology is an explicit specification of a conceptualization.” To add some more depth to this definition we also have from Tom,

“The term is borrowed from philosophy, where an ontology is a systematic account of Existence. For knowledge-based systems, what “exists” is exactly that which can be represented. When the knowledge of a domain is represented in a declarative formalism, the set of objects that can be represented is called a universe of discourse. This set of objects, and the describable relationships among them, are reflected in the representational vocabulary with which a knowledge-based program represents knowledge. Thus, we can describe the ontology of a program by defining a set of representational terms. In such an ontology, definitions associate the names of entities in the universe of discourse (e.g., classes, relations, functions, or other objects) with human-readable text describing what the names are meant to denote, and formal axioms that constrain the interpretation and well-formed use of these terms”¹

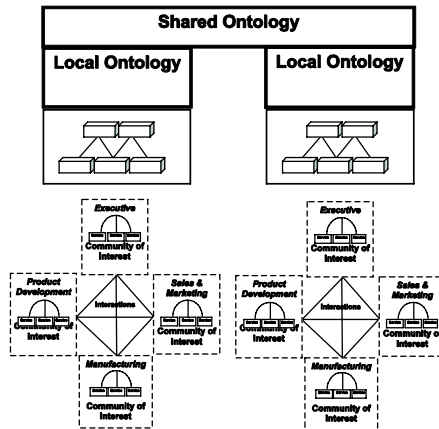
With this definition in hand, we can use it to make the claim that: given that Communities of Interest (COIs) exist, it is highly likely that the discourse or interactions within a given COI would represent concepts contextualized by that COI potentially resulting in an ontology specialized and specific to that COI. If this notion is taken a step further, the realization of the ontology as a human or machine readable construct would also be specialized and specific to that COI. And of course from the perspective of the people in that COI, the ontology would be deemed a “local” ontology.

If it existed, the local ontology would logically overlay on top of the computerized community constructs as shown below.

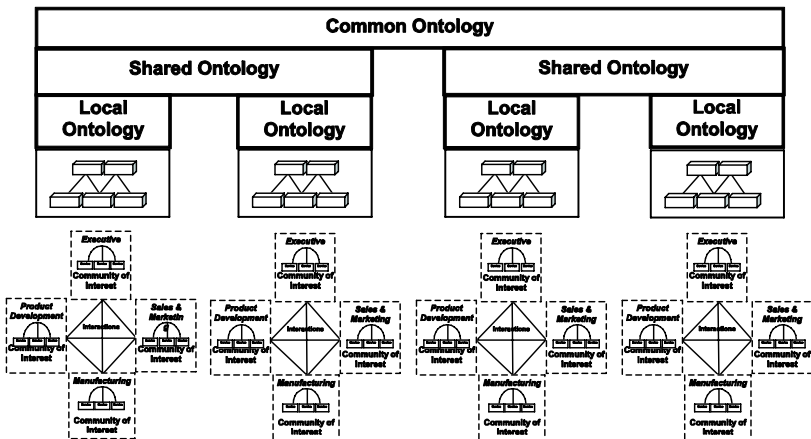


¹ “A translation approach to portable ontology specifications”, Tom Gruber, Knowledge Acquisition 5, 1993, pp. 199-220.

But companies or Communities of Interest do not usually work in isolation. Usually they are part of a larger value chain and as such need to share their information with their partners and in so doing create what is termed a “shared” ontology.



And of course, if there needs to be a reason to bring all concepts together to create the required and appropriate equivalencies and knowledge set for all things, we would end up with a “common” ontology layer, perhaps that which is envisioned for the World Wide Web.



3.2 The Methodology

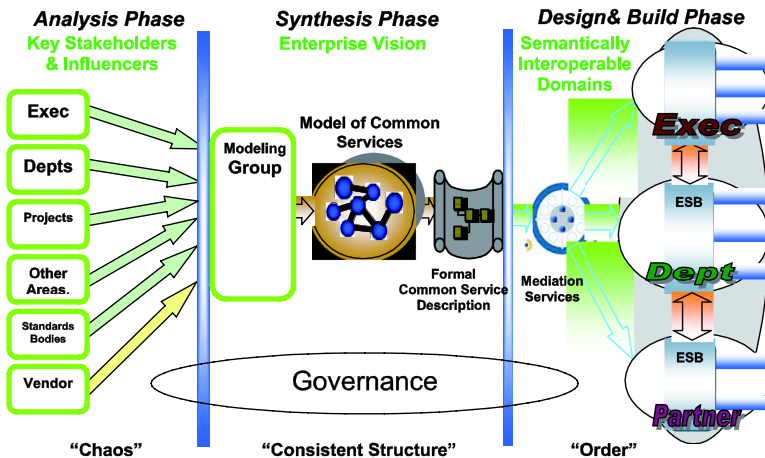
Overview

The methodology used to create a deployable common ontology is one based on known best practices. First an analysis phase is initiated whereby information is gathered and analyzed as to its structure and placement within the ontology. Concepts

from local and shared ontologies and knowledge from all areas are examined for their relevance (is anything not relevant in a common ontology?) and the concepts are classified within a taxonomy. There are millions of entities and influencers that affect the outcome of the analysis phase therefore rigour in choosing the most appropriate concepts must be applied.

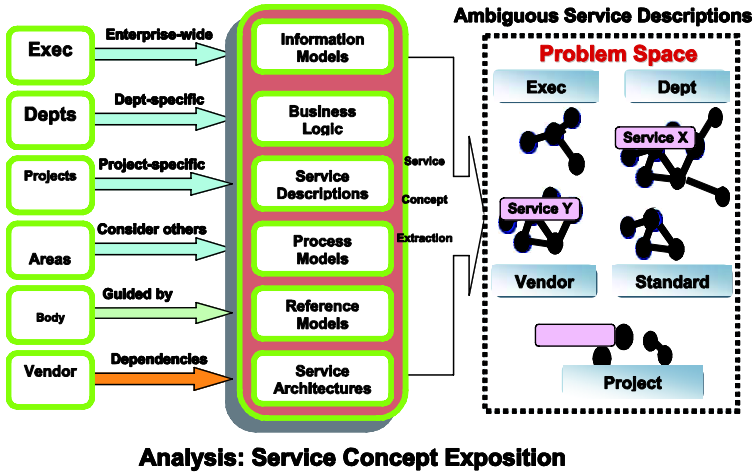
The second phase is the synthesis phase whereby all of the analytical output is captured in a modeling tool to formalize the structure and make it explicit. Tools such as Protégé, from Stanford University is one such tool that allows the creation of an ontology that can be viewed, edited and outputted in formats usable on post processing or model driven engines. This post-processing is required to create common service descriptions that are structured enough to be consumed within the automated world.

The last phase is the design and build phase whereby the common service descriptions generated in the second phase, which apply to the environment as a whole, are then captured, stored and processed through the Automated Mediation Capability (AMC). The AMC substitutes the generalized term which may have no meaning in a realized environment to a local syntax or semantic that has meaning within the context of the local community. This syntactic and semantic transformation allows a local community to invoke services from a remote community using a local syntax.



Analysis Phase

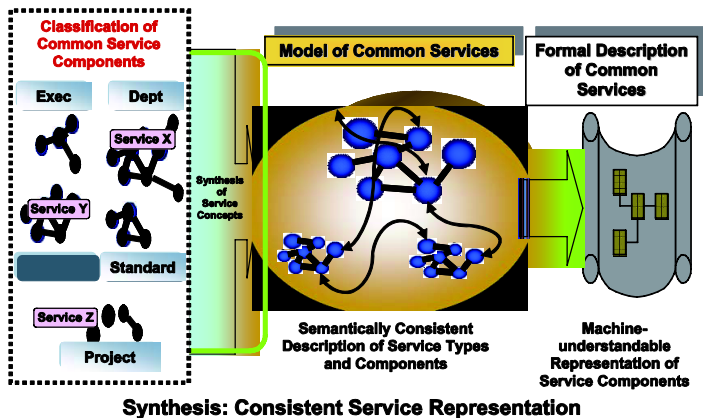
As discussed, the analysis phase is primarily concerned with exposing and discovering the underlying concepts, properties, attributes and meaning behind all of the relevant items in the environment that are to be captured within the system. A thorough examination of all of the entities through a scientific lens is required to determine relationships and understanding and most of all solicit agreement. The detailed methodology to support this phase is beyond the scope of this paper.



Synthesis Phase

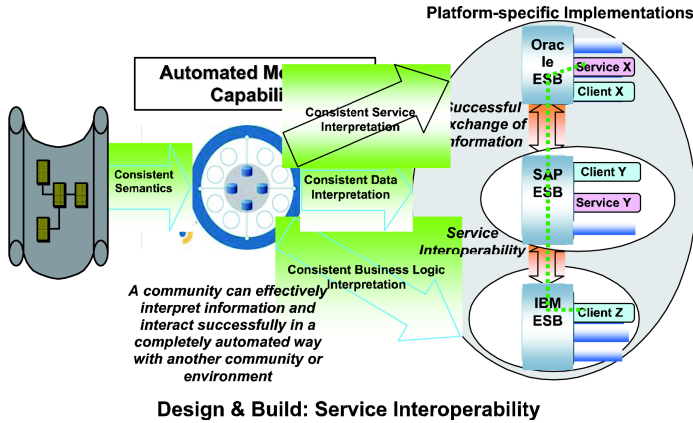
The subsequent synthesis phase allows the modelling group to create an overlay structure on top of the entities deemed appropriate for the system. This overlay structure is captured within a modelling tool such as Protégé from Stanford University or some other commercial package. Equivalencies between terms, unions and sets of entities that belong together in some way are modelled creating a complete as possible set of transformations and translations. Naturally, a set of taxonomies is required for classification of the entities and placement within certain compartments for ease of understanding and searching. This step results in the creation of the common ontology based on shared and local ontologies.

Preliminary formal common service descriptions are then constructed (through post modelling tooling) that provide the capabilities desired by the system. None of the formal service descriptions contain localized syntax or ontology although local or shared ontology may be promoted to the common layer if appropriate. These common service descriptions are in a machine interpretable format allowing automated processing in the last phase.



Design and Build Phase

In this last phase, the Automated Mediation Capability (AMC) provides the runtime translation from the common ontology to the local ontology. The localized ontologically correct service descriptions are then forwarded to the local ESB where they are registered in the local syntax. The registered service is now available for local service invocations but is also visible to the other communities through the AMC.



4 Issues

4.1 Overview

Knowledge is power. This is a well known phrase. If the knowledge of the system is only owned by one organization, how do partner organizations get changes into the analytical models in a harmonious way? We know changes to standards through standards bodies usually requires significant lead time, therefore the core team of modellers must be responsive to the needs of the ultimate end service providers and consumers. Bottom line: is the governance of the entire system a monarchy (with centralized control and all that this implies) or is it a democracy with policies such that all can make changes as long as those changes are well managed.

4.2 Run-Time Translation

There are advantages and disadvantages for centralized systems and there are advantages and disadvantages for distributed systems. If the model is centrally managed, does it make sense that the runtime mediation environment be distributed given today's high speed networks and compute engines that are available? Today, the movement is towards a "cloud" environment where location of function is independent of the consumer who uses it – essentially the world is a distributed place. How can a system profit from both models simultaneously? At this point, it is suggested that location of logic be located where best managed since human intervention and activity is the most costly resource. This area requires more research since the question of distribution is a critical issue.

4.3 Conflicts

Governance rears its ugly head in almost all areas of human endeavour. Who is in charge and who makes decisions is a source of great conflict. If the ownership issue discussed above has been answered, one would assume the issue of governance would also be solved. But conflicts do occur especially if the owners have a laissez-faire attitude towards conflicts between those of “lesser” means.

Some conflicts may be deployment based, some may be model based but in all cases, clear and sound practices must be adhered to, to minimize these effects.

4.4 Inconsistencies

In complex systems, and especially in systems where the entire problem space is not known going in, rules can be put in place that allow inconsistencies to be dealt with in an automated way. For example, should a term present itself in an ESB environment that is not known to the deployed mediation services, it can send it up to a learning program that updates the models and places the term or concept in the appropriate place in the model. These types of learning systems are complex, somewhat costly (if they exist) and probably are not commercially available since this is a burgeoning field.

4.5 Technical Interoperability

There are a number of technical specifications available that lay the foundation for the creation of deployable mediation services. Are they enough? Are they going to change? Who uses them? What if you don't want to use them? What if they don't meet the needs of the system?

It is important to note that knowledge engines and the technology that supports them progresses at a certain pace and the realization of dynamic mediation can only be as good as the technology which is available at the time. Even though science fiction shows do exist and illustrate things which could be, we need to temper our enthusiasm with the knowledge of how things really are.

4.6 Semantic Interoperability

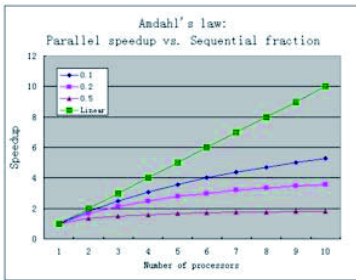
Terms and concepts only have meaning to humans. The notion that computers “understand” things is delimited by their ability to process information via a given algorithm. True computerized thought is not yet possible. Given this, we know that manual effort is required to model concepts and create the relationships within a particular model that then can be automated.

In today's economy, companies and institutes of higher learning are aware of the cost of endeavours which may or may not yield anything of commercial value. Knowledge for knowledge sake has its place, but we are now, more than ever, driven by investors and the holders of the “cash” then we used to be, and if this is so, how far can we go with creating the common ontology where all services can connect and interoperate through a framework of semantic transference?

5 Implications

One of the great implications of an ontological overlay, in this author’s opinion, is the reduction in complexity in system design. The complexity of the system (and the knowledge contained within it) are kept at the top most layer, the model and ontology itself, and the working infrastructure has its runtime orders “pushed” into it requiring no “hard-coding”. This of course implies that the execution platform is extremely flexible, dynamic, responsive and capable of learning. For the purposes of this paper, the existence of such a runtime platform is assumed and the description of it is beyond the scope of this paper.

Turning to the notion of complexity using Amdahl’s law as a basis, we have:



Amdahl's law, (Gene Amdahl, 1967)

Named after computer architect Gene Amdahl, is used to find the maximum expected improvement to an overall system when only part of the system is improved. It is often used in parallel computing to predict the theoretical maximum speedup using multiple processors.

Source: www.wikipedia.com

If **S** is the fraction of a calculation that is serial and **(1-S)** the fraction that can be parallelized, then the greatest speedup that can be achieved using **P** processors is:

$$\frac{1}{(S + (1 - S) / P)}$$

which has a limiting value of **1/S** for an infinite number of processors.

Source: www.phy.duke.edu

Clearly, using multiple processors and distributing subtasks to them to parallelize computation is known and proven. Knowing this we can easily extend this idea and substitute service invocations and orchestrations for linear algorithmic subtasks.

Service orchestrations can have many levels and can be quite broad. There really is no limit to their extent, therefore the parallelism originally described by Amdahl today can be thought of as parallelism in more than one dimension. This n-dimensional parallelism can increase overall performance n-fold.

Extending this idea further, if the orchestrated services are well-formed, meaning they use a consistent model, semantic and syntax, is not the job of programming eased through the use of models and code reuse? Furthermore, if consistency in design is inherent within all aspects of the system, have we not also reduced the cumulative complexity of its creation, enhancement, management, and deployment? The generalized pattern for system creation could be used over and over and all that has to be changed for a new instance is the model that drives it.

So at this point we have a faster and less complex system to create system instances from as required. These system instances now based on a common design and deployment model can be easily interconnected and since the ontologies both local and shared can be linked into the common ontology, the discontinuity between the system instances can be dissolved.

Simpler and consistently designed and built machines as we have seen time and time again are those that get used the most. Propagation throughout the world of very complex systems is very rare. For example, Large Hadron Colliders (LHCs) are not built every day since they are very complex one-off devices whereas mobile phones are built and used around the world every second.

Continuing the mobile phone example, the mobile phone is consistent in design, uses interoperable protocols and is built on similar and consistent designs and platforms. The mobile phone is a classic example of removing discontinuity of an environment through the reduction of complexity. People communicate freely (discontinuity reduction) via the simplicity of the handheld device (reduction of complexity). The mobile phone hides and encapsulates the complexity of the models and the technology that it is built from. The ontological system described herein, hides and encapsulates the complexity of interaction in a similar fashion.

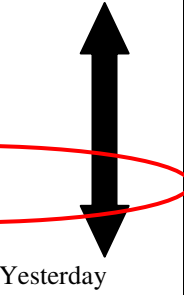
6 Final Considerations

One of the more interesting aspects regarding the notion of a common ontology and its usage in a service oriented environment is how it maps to a psychological view of human behaviour. Abraham Maslow, in 1943 described the “Hierarchy of Needs” which listed an order of needs that are satisfied by humans with respect to the environment, other humans and the interactions between them. It is interesting to put a computer spin on this as this author believes there is a definite hierarchy of computer-based services supported by a common ontology that also maps well to the Maslow model.

This paper started its discourse on the Community of Interest model. It then proceeded to link communities together into a holistic perspective through the common ontology with a potential end-to-end solution (albeit at a very high level) to provide a foundation for higher-order automation. But the discussion need not end there. By creating a “smart” or even an “aware” system that self-corrects and learns from new terms and their context implies this system is definitely heading up Maslow’s hierarchy. For example, infrastructures and their containing enterprises map well to level one and level two of Maslow’s hierarchy respectively. The ESBs in support of a Community of Interest provide the interaction at the community level satisfying the third level of Maslow’s hierarchy. The forth level is a very human need for self-esteem and distinction within a group. This author is not aware of any computer that needs to be “happy” or differentiate itself within a group.

Looking at level five, a lot of the Artificial Intelligence (AI) work is definitely progressing in this area. The main issues are creating models and software emulating what happens in the analysis and synthesis phase into algorithmic approaches that satisfy all contexts. The human mind excels at the “Wisdom and Reasoning” layer and adapts with all of the ability that it can muster and systems have a ways to go before being programmed accordingly to learn and reason.

Examining the top layer of Maslow’s hierarchy, if a system is fully realized in layer 5, it should be able to be adapt, even self-perpetuate and if proper rules and reasoning engines are incorporated into it then without a doubt it can be autonomic. The level of self-actualization attained is truly dependent upon investment by the development community and the state of technological advancement.

Maslow		Interpretation	
Need	Notional	Service Types	Time frame
6 – Self Actualization (id, ego, superego)	Abstraction & Continuity	Adaptive, Autonomic & Self-Perpetuating	
5 – Aesthetic & Cognitive (knowledge, understanding, goodness, justice, beauty, order)	Wisdom & Reasoning	Knowledge	
4 – Esteem (competence, approval, recognition)	Distinction	Differentiation	
3 – Belongingness & Love (affiliation, acceptance, affection)	Community	Community of Interest	
2 – Safety (security, physiological safety)	Growth	Enterprise	
1 – Physiological (food, drink air)	Survival	Infrastructure	

7 Conclusion

The movement to a service oriented view of the world is definitely taking place. The understanding that man’s interaction with man, and man’s interaction with nature is service based has become pervasive. But knowing this fact is not enough. If we are to progress technologically where more and more tasks are automated, there has to be a way to “connect” things together without expending huge efforts in doing so. Most of the time spent learning and educating is about transference of meaning. A student may hear a term and recognize it (syntactic awareness), but only when they understand it has the knowledge been transferred (semantic awareness).

For the automated marketplace, the above holds true as well. If companies or organizations are to exchange goods and services in an automated way, the meaning of things has to be part of the exchange itself and not performed separately as part of a human or out-of-band negotiation. Therefore as mankind moves from the Community of Interest model of service exchange to the commoditization of services in the general marketplace, this movement will necessitate the creation of a generalized ontological overlay as part the foundation of instantaneous service provision and consumption.

A Non-technical User-Oriented Display Notation for XACML Conditions

Bernard Stepien, Amy Felty, and Stan Matwin

School of Information Technology and Engineering, University of Ottawa, Canada
and Devera Logic, Inc., Ottawa, Canada

{bernard.stepien, amy.felty, stan.matwin}@deveralogic.com

Abstract. Ideally, access control to resources in complex IT systems ought to be handled by business decision makers who own a given resource (e.g., the pay and benefits section of an organization should decide and manage the access rules to the payroll system). To make this happen, the security and database communities need to develop vendor-independent access management tools, useable by decision makers, rather than technical personnel detached from a given business function. We have developed and implemented such tool, based on XACML. The XACML is an important emerging tool for managing complex access control applications. As a formal notation, based on an XML schema representing the grammar of a given application, XACML is precise and non-ambiguous. But this very property puts it out of reach of non-technical users. We propose a new notation for displaying and editing XACML rules that is independent of XML, and we develop an editor for it. Our notation combines a tree representation of logical expressions with an accessible natural language layer. Our early experience indicates that such rules can be grasped by non-technical users wishing to develop and control rules for accessing their own resources.

Keywords: Access control, notation, rule editor, XACML.

1 Motivation

The XACML (eXtensible Access Control Markup Language) [5] access control language (ACL) is naturally precise since it is based on an XML schema that represents the grammar of a given application. But this very property puts it out of reach of non-technical, and especially casual users that in some cases could even be computer illiterate. The main obstacles for a casual user in using XACML are:

- Long XML tags;
- Long and complex domain references;
- Prefix notation for operations;
- List-oriented notation for conjunction and disjunction operators.

While it is practically impossible for a casual user to start coding rules with a text editor—this would require full knowledge of XML and XACML grammars—a first step toward solving this problem could be to use an XML editor that frees the user from this

knowledge up to a certain point, as the supplied XML Schema enables the selection of appropriate tags in a context-oriented way.

A number of such tools exist in different syntaxes and formats, each trying to address a specific technical problem. They can be classified into two broad categories:

- Generic XML editors;
- Specialized application oriented XML editors—where XACML belongs.

While all of these editors claim to be targeting non-technical users, their documentation indicates that they require at least a basic knowledge of XML. In fact, one of the main problems with the XACML notation is that it requires some programming skill regardless of the tools used.

Currently, there is a very limited set of XACML tools. The UMU editor [3] was the first attempt to have a general XACML editor. Others have further refined the specialization. This is the case of the visual Language hierarchy solution [2] that exclusively targets RBAC [6] applications.

Our new approach has been guided mostly by the study of existing editors. There are a number of open source and commercial XML and XACML editors available that follow a number of basic principles.

2 Principles in Current XML Editors

XML editors are most often based on a tree display principle of an XML document. The tree display is most natural, mostly because an XML document is hierarchical by definition.

XMLPad [8] is the most commonly used open source editor. It offers three different views of an XML document: the XML plain text, the grid and the table view. In addition to these views, a document outline represented as a tree is also available.

Let us imagine that we need to create a rule that authorizes a purchase action if a specific condition holds. Let us use a simple condition that says that a purchase is

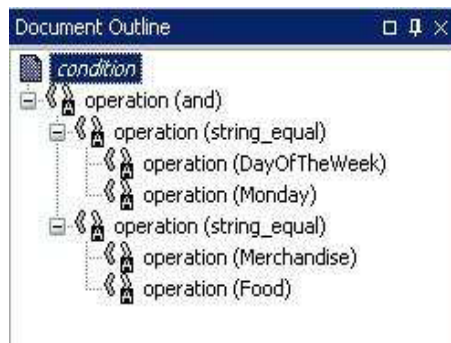


Fig. 1. Document outline of a simple condition

Fig. 2. XML source of the condition shown in Fig. 1

Fig. 3. Grid view of the condition whose XML is in Fig. 2

permitted if the day is Monday and the merchandise purchased is food. This condition would have a document outline as shown in Fig. 1. Such an outline mainly shows the name of the node and the value.

The corresponding XML source view is shown in Fig. 2. It can be interactively edited by positioning the cursor in a region, which triggers the appearance of a choice of actions. Examples of actions include entering the value of a new attribute if it is not already present, or appending a new tag. The editor will automatically insert the attribute or tag selected from a drop down menu. Thus here, the interesting principle is that

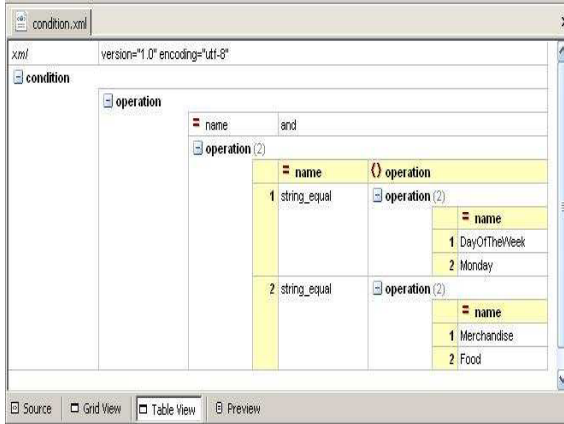


Fig. 4. Table view of the tree shown in grid view in Fig. 3

although the user sees only plain text, the editor provides features that waive the need for in-depth knowledge of the data model (DTD or Schema) and thus reduce the risk of errors such as spelling mistakes of attribute names or forgetting an attribute altogether. The source view however allows the direct typing of tags and attributes and a parser is triggered at every save attempt and highlights errors.

The corresponding Grid view is shown in Fig. 3. It corresponds to a horizontal tree where each node indicates the tag names and their corresponding attributes and also the related DTD for the current element. Again, features similar to those available in the source view are also available. Here however, the presentation of the data model could actually assist the user in planning his next move.

The table view shown in Fig. 4 is just another way to represent the tree of the grid view, attempting to further reduce the programming skills required of the user. Note also the attempt to reduce the amount of information in the tree by factoring out the name of the tag when there are multiple occurrences of a tag, as in this example for the arguments of an operation.

3 Principles in Current XACML Editors

In order to understand the implications of writing a XACML specification of the previous simple example, we need to examine the representation of the condition of this example in XACML.

```
<Condition FunctionId=
  "urn:oasis:names:tc:xacml:1.0:function:and">
  <Apply FunctionId=
    "urn:oasis:names:tc:xacml:1.0:function:string-equal">
    <Apply FunctionId="urn:oasis:names:tc:xacml:1.0:
      function:string-one-and-only">
```

```

<SubjectAttributeDesignator AttributeId="Merchandise"
  DataType="http://www.w3.org/2001/XMLSchema#string" />
</Apply>
<AttributeValue DataType=
  "http://www.w3.org/2001/XMLSchema#string">food
</AttributeValue>
</Apply>
<Apply FunctionId=
  "urn:oasis:names:tc:xacml:1.0:function:string-equal">
<Apply FunctionId="urn:oasis:names:tc:xacml:1.0:
  function:string-one-and-only">
  <SubjectAttributeDesignator AttributeId="DayOfTheWeek"
    DataType="http://www.w3.org/2001/XMLSchema#string" />
</Apply>
<AttributeValue DataType=
  "http://www.w3.org/2001/XMLSchema#string">Monday
</AttributeValue>
</Apply>
</Condition>

```

The first XACML editor, developed by University of Murcia [3] is shown in Fig. 5. It is based on two complementary views, one for the document outline and one for the attribute values and some local overviews.

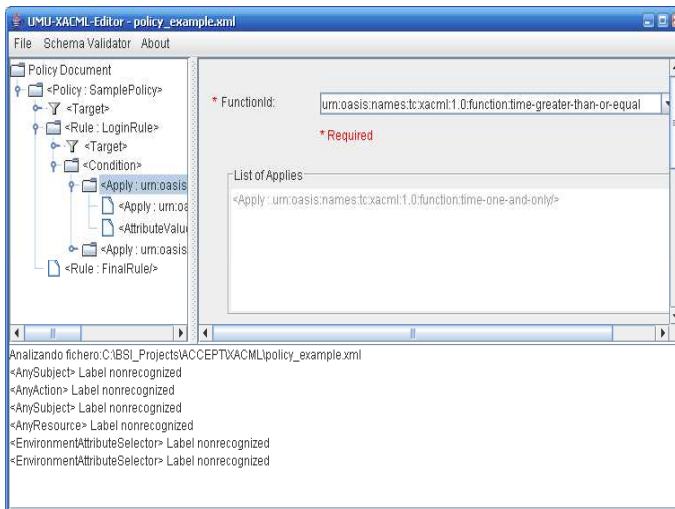


Fig. 5. UMU editor representation of the condition

The first problem this editor has addressed is omitting the need to type the domain names. Functions are merely selected from lists along with their domains.

Conditions are constructed by clicking on a node of the tree and selecting an operator from a list. Again, while this editor reduces XACML coding efforts considerably, it requires a strong expertise both in XML and XACML.

This editor is not easily usable by a non-technical user, mostly because this kind of user will not know the XACML condition grammar. Also the resulting tree is again reorganizing the terms of a condition in a way that is not mapped directly on to the corresponding natural language statement of the condition. For example the *and* operator is located at the top of the tree hierarchy instead of being in the middle.

More recently the XACML Studio editor tried to alleviate some of the difficulties of use mentioned about the UMU editor [7] but with most of the same functionalities.

One principle is important in both general purpose and specialized editors presented so far. All editors provide the capability to hide or expand portions of the tree in their various views except the source view. This feature allows the user to focus on a portion of the tree and thus avoids the cluttering that naturally results from the presentation of large amounts of information. This feature has, however, an important side effect. It prevents the user from having an overview of the entire condition he is trying to assemble. This makes the reasoning about the logic of the expression being built very difficult and could lead to errors.

4 Our Proposed Notation

Our proposed notation is only a display notation. It is neither a new language nor a replacement for XACML. However, it bears some formal qualities that we have chosen mostly to facilitate its use in interactive editors that allow a non-technical user either to create a new policy or to modify an existing one. Effectively, if we had followed only the consideration of making the policies and rules understandable by a non-technical user we could have merely translated them into plain English, but we quickly realized that plain English would have been a challenge to manipulate in an editor. Instead we use trees to represent logical expressions. Knowing that a casual user may not grasp abstract mathematical concepts, we decided to create a hybrid between formalism and plain English. This idea has previously been mentioned as a goal for the XACML community by Vullings [4], but we are aware of no published paper on research results in this area.

Furthermore, we came to the conclusion that a full non-technical representation of XACML is not really possible, mostly because XACML is a strongly typed language. Typing is not a concept that the casual user can grasp beyond the basic types, like numeric or alpha-numeric. Effectively, the nuances of data storage considerations that further divide numeric types into various levels of precision such as integer, float, double, etc. can only be correctly manipulated by technical users. However, the actual display of a XACML condition has no real barrier of this kind, and can be considered user-friendly.

Consequently, we propose a separation of concerns between the data typing definitions that should remain in the hands of knowledgeable IT technicians, and the policy editing including its logical expression construction that can be delegated to the non-technical user.

This approach is appropriate mostly because an access control application is available within a context where there is an infrastructure organized by the provider of the service. This infrastructure naturally includes the definition of variables along with their

types and potential allowed values. For example, an eStore will define what products it will sell along with the necessary parameters such as product identifiers or codes, unit types to express their quantities ordered, etc.

Data typing is thus relegated to another document that we also decided to structure using XML, where variables used in a given application are defined along with their data types and potential lists of allowed values.

Our notation is based on the following basic principles:

- Stay as close as possible to the user’s natural language by avoiding any technical terminology for operators and maintaining the overall structure of a natural language.
- Offer an implicit structuring by organizing the natural language into a tree.
- Organize the tree so as to make it consistent with the natural language statement of the condition by using an infix representation for conjunction and disjunction operators.
- Maintain XACML’s natural non-binary nature of conjunction and disjunction operators but eliminate its original list representation.
- Use a different, yet still casual terminology for conjunction and disjunction operators depending on their position in the tree hierarchy.
- Ensure a full graphical overview of the expression being built at all times regardless of its complexity. This implies no capability to collapse portions of the tree.

Thus our notation is very close to a natural language statement of the condition. It is actually an improvement over it, as it shows the logical structure of the condition. This will prove very important when building complex expressions requiring the concept of operator precedence. A casual user should not have to be concerned with representing operator precedence.

Our previous example augmented with an additional conjunction would be represented in our notation as follows:

```
DayOfTheWeek is Monday
and
Merchandise is Food
and
BalanceOfAccount over 500
```

The simple example above has a very shallow depth. Two additional techniques can be used to express more complex conditions:

- Allowing multiple values for a given variable;
- Allowing sub-constraints on a value for a variable.

The first principle is illustrated in the next example where the condition is extended to two different days of the week and to two different kinds of merchandise:

```
DayOfTheWeek is one of Monday, Friday
and
Merchandise is one of Food, Travel
and
BalanceOfAccount over 500
```

The above example also illustrates that our notation is not relying on a one-to-one mapping to XACML. For example, in our notation we show only one occurrence of the variable name `DayOfTheWeek`. In XACML this would be represented instead by a disjunction operation on two sub-expressions of the kind “`DayOfTheWeek` is Monday or `DayOfTheWeek` is Tuesday” where both disjuncts use the XACML *string_equal* operator. However, when the user saves this expression, it is translated to XACML syntax where the variable is repeated for each subexpression.

The second principle is illustrated by introducing sub-constraints on values by saying that travel is allowed only on Friday and food purchases only on Monday or Tuesday. Here, the conjunction operator *and* has been represented by the phrase *provided that* which is more natural since it is in the context of a disjunction.

```

Merchandise is one of
  Food
    Provided that DayOfTheWeek is one of Monday, Tuesday
  Travel
    provided that DayOfTheWeek is Friday
and
  BalanceOfAccount over 500

```

The above expression corresponds to the following plain natural language version: “It is permitted to purchase food on a Monday or a Tuesday or travel on a Friday provided that the balance of the account is over 500.”

As we can see from this example, the order of the sub-constraint in the pure natural language version is strictly the same as in our notation. The only difference is the graphical structuring of the tree appearance. It helps clarify the rule in its natural language form, where putting various sub-constraints in their appropriate context requires mental effort from the user.

Another advantage of the tree notation we are proposing is that it avoids the ambiguity of the scope of the disjunction operators. In the natural language representation above it is hard to understand the exact scope of the *or* operator that applies to food or travel because of the presence of the other disjunction about Monday or Tuesday. In our tree like notation this ambiguity disappears entirely. It is a well known fact that this kind of scoping problem is the prime source of ambiguities in interpreting statements in natural language. In fact, with traditional non-XACML notation for logical expressions, the only way to resolve these ambiguities would be to use parentheses as follows:

```

(((Merchandise == Food) and
  ((DayOfTheWeek == Monday) or (DayOfTheWeek == Tuesday))) or
  ((Merchandise == Travel) and (DayOfTheWeek == Friday))) and
(BalanceOfAccount > 500)

```

Note that since our notation is only for display purposes, it is not meant to be parsed and thus no grammar needs to be defined for it. In fact, the editor maintains an internal mapping between pure XACML and the version displayed using our notation. One of the main features of this notation is that we do not translate conjunction and disjunction operators to a single phrase. For example, a conjunction is represented either with the word *and* or paraphrased with an expression such as *provided that*. The latter implies

conjunction but is more conceptually precise when it appears in the scope of a disjunction, naturally resolving the ambiguities that a mix of conjunction and disjunction operators unavoidably yields. We further resolve such ambiguities using indentation.

Another consideration is that we do not intend to cover the entire set of capabilities provided by the XACML grammar. This is mostly due to the fact that, as pointed out already, in recommended use of XACML, logical expressions should remain simple so that they remain understandable, since complex expressions in XACML can be extremely hard to read. On the other hand, our notation allows users to compose complex logical expressions that remain readable and understandable, and thus may call for use of the full XACML grammar.

We also support the XACML negation operator by merely integrating it in the natural language representation as follows:

```
Merchandise is not Food
```

We also support XACML variables to abbreviate a portion of a logical expression. For example, a variable can be introduced to represent week days. The abbreviated expression either would be a disjunction between equalities, one for each day, or would use a member-of construct provided by XACML.

5 Our Notation in the Context of an Editor

We have developed a XACML editor as a series of interfaces in which our notation is used in all cases where an expression is required such as in target subjects, resources and action specifications, and in the conditions of rules.

Our XACML editor reads a configuration file which specifies the names, data type and potentially allowed values from an XML file as in the following example:

```
<Variable name="DayOfTheWeek" type="String">
  <Values>
    <Value name="Monday" />
    <Value name="Tuesday" />
    <Value name="Wednesday" />
    <Value name="Thursday" />
    <Value name="Friday" />
    <Value name="Saturday" />
    <Value name="Sunday" />
  </Values>
</Variable>
```

The XACML policy interface allows the user to create or modify a policy. The rule interface allows creating or modifying a rule and especially its condition, as shown in Fig. 6.

A modification is achieved by first double clicking a word in a condition and then invoking the requested modification by clicking one of the tool bar buttons, which allow operations such as modifying a value, adding, modifying or deleting a constraint or inserting an additional value. The insertion or modification of a value is achieved via a value selection interface show in Fig. 7. In Fig. 6 clicking the value food is sufficient to



Fig. 6. Our XACML Policy Interface



Fig. 7. Policy value modification using our editor

obtain all the possible values of the Merchandise variable. The internal representation, which is a tree that is mapped exactly onto the XACML structure, enables the editor to determine which variable a clicked value corresponds to, and thus provide the appropriate value selection interface. A value node is a leaf of an operation node such

as *string_equal*. Walking the tree to the parent of the value and then descending from the parent to the leaf that contains the variable makes this process possible. Once the appropriate selection is done in the value selection interface of Fig. 7, the resulting tree is redrawn along with all the internal references to type definitions.

6 Our Notation Beyond XACML

While our efforts have concentrated on XACML, we have applied the same principles to other access control languages such as Cisco IOS [11]. This has been made particularly easy by the architecture of our editor, where the internal representation of a policy is independent of the XACML language itself. Our internal representation, however, provides the structure of XACML, but without reference to its tag names or types. Thus the XACML language structure is used as a common denominator for handling all other access control languages. Our editor has a policy connector component that can handle an unlimited number of languages provided that parsers for these languages are built. Another benefit of this language-independent internal representation is that the editor can be used to translate one language into another language. This requires adding the appropriate code generators that all operate on the language-agnostic internal representation.

The following example shows a Cisco IOS rule and its corresponding representation in our notation. The variable names are defined by the translator as they are not part of the original syntax of Cisco IOS. The rule:

```
access-list 101 deny tcp host 148.22.33.44 host 192.168.0.0
                    eq 3500
```

is displayed in our notation as follows:

```
    protocol is tcp
and
    srcIP is 148.22.33.44
and
    dstIP is 192.168.0.0
and
    dstPort is 3500
```

7 Conclusion

XACML editors can be an effective and highly desirable tool, assisting non-technical users in specifying complex XACML rules, e.g., for access and resource control. We have proposed here a simple yet powerful, implemented notation that allows users to perform this task by providing a representation that is very close to natural language. Also, due to its high compactness, it provides a rare overview quality that is an important factor in reducing errors, thus helping to ensure the commercial success of the application.

Our early experience with several non-technical users confirms that our goal of empowering non-technical users with a tool giving them control of their resources can be

met with the proposed notation. We need to perform a more thorough evaluation of how well this goal is realized, and collect more experience in representing a variety of resource access specifications using the approach and the editor described in this paper.

Our editor based on our notation is not intended to be a replacement for any XACML editor when the user is fully technically qualified. However, while our initial goal was to address the needs of casual, non-technical users, an additional benefit of this approach is that even technical users can easily specify very complex conditions, something that was stated as important to avoid in the past in the XACML user community. This has an important consequence of avoiding the splitting of complex rules into numerous rules with narrower targets, which produces large rule bases that become rapidly unmanageable.

References

1. Boney, J.: Cisco IOS in a nutshell, 1st edn. O'Reilly, Sebastopol (2001)
2. Giordano, M., Polese, G., Scanniello, G., Tortora, G.: Visual Modelling of Role-Based Security Policies in Distributed Multimedia Applications. In: 6th IEEE International Symposium on Multimedia Software Engineering. IEEE Press, Los Alamitos (2004)
3. University of Murcia XACML Policy Editor,
<http://xacml.dif.um.es/>
4. Vullings, E.: Implementing Authorized Access (2006),
http://www.apsr.edu.au/Open_Repositories_2006/erik_vullings.ppt#256
5. XACML, OASIS standard,
http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml
6. XACML Profile for Role Based Access Control (RBAC) (2004),
<http://docs.oasis-open.org/xacml/cd-xacml-rbac-profile-01.pdf>
7. XACML Studio, <http://xacml-studio.sourceforge.net/>
8. XMLPad, open source, <http://www.wmhelp.com/xmlpad3.htm>

Goal-Driven Development of a Patient Surveillance Application for Improving Patient Safety

Saeed Ahmadi Behnam¹, Daniel Amyot¹, Alan J. Forster², Liam Peyton¹,
and Azalia Shamsaei¹

¹ SITE, University of Ottawa, 800 King Edward Ave

² Department of Medicine, University of Ottawa and OHRI
Ottawa ON, K1N 6N5 Canada

sahma088@uottawa.ca, damyot@site.uottawa.ca, aforster@ohri.ca,
lpeyton@site.uottawa.ca, asham092@uottawa.ca

Abstract. Hospitals strive to improve the safety of their patients. Yet, every year, thousands of patients suffer from adverse events, which are defined as undesirable outcomes caused by health care business processes. There are few tools supporting adverse event detection and these are ineffective. There is hence some urgency in developing such a tool in a way that complies with the organizations goals and privacy legislation. In addition, governments will soon require hospitals to report on adverse events. In this paper, we will show how a pilot application we developed contributes to the patient safety goals of a major teaching hospital and how our goal-driven approach supported the collaboration between the university researchers and hospital decision makers involved. Benefits and challenges related to the modeling of requirements, goals, and processes, and to the development of the application itself, are also discussed.

Keywords: Adverse Events, Business Process Modeling, Goal Modeling, Health Care, Patient Safety, User Requirements Notation.

1 Introduction

Modern health care is a data- and knowledge-intensive enterprise. Information technology (IT) systems are increasingly used in health care organizations to collect, analyze, manage, and share information and knowledge. Although one of the main goals in this industry is to improve quality of care, IT systems are often not aligned with this primary goal. According to a recent report from the US National Research Council [11], in which the authors studied eight medical centers acknowledged as leaders in their usage of IT, such systems in health care are used in practice more for regulatory compliance and lawsuits protection than to improve clinical care.

Patient safety is one important sub-goal of health care quality, and minimizing the number and severity of *adverse events*, which are undesirable patient outcomes caused by medical care, contributes greatly to patient safety. It is not only important for hospitals and other health care organizations to define and support processes for detecting, assessing, and reporting on adverse events, but, in fact, this is being turned into a legal obligation in many provinces and states.

Often, paper-based approaches are used to support such processes, and they may vary from department to department. In this context, there is both a need and an opportunity to take advantage of e-technologies to improve the efficiency and effectiveness of existing health care processes. However, it has been observed that current IT applications in this area tend to “simply mimic existing paper-based forms and provide little support for the cognitive tasks of clinicians or the workflow of the people who must actually use the system” [11].

This paper reports on our experience and lessons learned during the development of a *Patient Surveillance* application targeting the detection of adverse events. This project is a joint venture between health care professionals from The Ottawa Hospital (TOH) and researchers from the Ottawa Hospital Research Institute (OHRI) and the University of Ottawa. This tool supports a prospective surveillance process in order to improve the accuracy of adverse event detection (and hence improve patient safety) while minimizing its associated costs.

The development approach taken is driven by the goals of the organization and other stakeholders, in order to avoid repeating the same mistakes identified for existing IT systems as discussed above. It combines state-of-the-art requirements engineering techniques and e-technologies. Requirements (e.g., goals, processes and database schemas) are elicited using a combination of models in the User Requirements Notation (URN) and UML. The main project objective, improving patient safety, was decomposed into four sub-goals: data collection, information generation, knowledge creation and knowledge application. Goal and process models were created for all of them, but the scope of the first phase of this project was limited to the first two goals.

A Web-based application was created and then used by a nurse to monitor patients using a mobile tablet PC for a one-month period (so far), and by physicians to assess whether the observations were indeed adverse events, with probable causes.

The rest of the paper is as follows. Section 2 provides background information on adverse events and on the notation used to model the target business process and its goals. In section 3, we describe our development approach, which is then detailed with the business process and the implementation of the surveillance application itself in section 4. Observations and lessons learned are discussed along the way. Finally, the last section provides conclusions and opportunities for future work.

2 Background

2.1 Adverse Events in Health Care

Adverse events are undesirable patient outcomes caused by medical care rather than the underlying disease process [12]. An example of an adverse event is an allergic reaction caused by a medication. The reaction would not have occurred if the patient had not been exposed to the medication. In most instances such events are not avoidable. However, in a substantial proportion, they are preventable as they are due to an error. For example, if the prescribing physician neglected to enquire about prior allergic reactions to medications when she prescribed the medication, then the patient may be exposed to a harmful medication unnecessarily.

Unfortunately, there are large numbers of adverse events in the health care system. Focusing specifically on hospital patients, Canadian studies estimate that one in twelve hospital patients experience an adverse event [2]. A third of these adverse events are preventable. More importantly, one in six patients dies as a result of the adverse events. Extrapolating this risk to the Canadian population of hospital patients, there are 28,000 deaths annually due to medical errors. While this statistic is alarming, the risks are probably greater across the entire health system, which includes institutional care and ambulatory care. Both of these settings are also associated with an important risk of preventable injury [5, 6].

The current health care industry has immature systems to detect and monitor adverse event occurrence. The accepted method of adverse event detection is voluntary incident reporting. The method does not identify over 90% of adverse events [6]. Despite this fact, it is mandated by most accrediting bodies. More sophisticated methods of adverse event detection have been tested and are in development, including two-stage chart review, administrative data surveillance, electronic health record surveillance, and clinical surveillance [9].

Prospective adverse event surveillance holds promise as method [4, 9]. In this approach, a nurse monitors patient care for pre-specified triggers. When they occur, specific information is recorded and a case summary is generated. Case summaries are reviewed by physicians to determine their importance. This is a very cost-effective approach, even when considering the nurse's salary. The approach was developed based on prospective surveillance experience in 5 hospital units. These units included: a general medical unit, an intensive care unit, a cardiac surgical intensive care unit, an obstetrical unit, and an orthopedic surgery unit. The general approach for identifying adverse events worked effectively in all units despite there being very different patients, work processes, and adverse event types identified.







Although this method identifies more adverse events than other techniques, there is a need to develop IT infrastructure to support its activities. Because the general approach is modified slightly for each unit as different patient characteristics are measured and different adverse event triggers are monitored, supporting software must be built to accommodate customization.






2.2 Business Process Modeling with the User Requirement Notation

The *User Requirements Notation* (URN) is a graphical modeling language recently standardized by the International Telecommunication Union [7]. URN is intended for the elicitation, analysis, specification, and validation of requirements. URN allows software and requirements engineers to discover and specify requirements for a proposed system or an evolving system, and analyse such requirements for correctness and completeness.

The applicability of URN goes beyond requirements models; URN is also suitable for the modeling and analysis of business goals and processes [10, 14]. URN is composed of two sub-notations: the *Goal-Oriented Requirement Language* (GRL) for goal modeling and *Use Case Maps* (UCM) for scenario/process modeling.

GRL enables business analysts and IT architects to model strategic goals and concerns using various types of intentional elements and relationships, as well as their stakeholders called actors (○). Core intentional elements include goals (◻) for

functional requirements, softgoals () for qualities and non-functional requirements, and tasks () for activities and alternative solutions. Intentional elements can also be linked by AND/OR decomposition and by contributions. Positive contribution levels may be sufficient () , insufficient () or some positive () . Similar levels exist for negative contributions () . Quantitative contributions on a [-100, 100] scale may also be used. GRL *strategies* enable modelers to assign initial satisfaction values to some of the intentional elements (usually alternatives at the bottom of the graph) and propagate this information to the other elements through the decompositions and contribution links. This ultimately helps assess the impact of alternative solutions on high-level goals of the stakeholders involved. Such models are also useful for evaluating trade-offs, documenting rationales for design decisions, and modeling legal requirements.

Use Case Maps (UCM) are used to model scenarios and processes in the form of causal relationships linking responsibilities () which may be assigned to components () . Responsibilities represent activities performed in a process whereas components represent actors, systems, and system parts. UCM support most of the concepts used in common workflow modeling notations including start points () , end points () as well as alternative and concurrent flows. Stubs () are containers for sub-maps and can be used to organize a complex model in a hierarchical structure.

In our project, we used the jUCMNav open source software, an Eclipse plug-in used for creating, analyzing, and managing URN models [13]. This tool also supports extensions to URN for modeling key performance indicators (KPI) in the context of business process analysis and monitoring and performance management [10].

3 Highlights of the Development Approach

The approach selected is described below. The goal and scenario modeling part is inspired from the process proposed by Liu and Yu [8]. Several micro and macro-iterations were performed along the way.

1. **Stakeholder and goals:** Model, with GRL and jUCMNav, the stakeholders and their main high-level goal. Decompose the goals of the main stakeholder, namely The Ottawa Hospital (TOH) in our project.
2. **Alternatives and strategies:** Model the alternative surveillance methods as tasks and their contributions to TOH's goals. The comparison among these methods, enabled by computing the results of GRL strategies (automatically done with jUCMNav), was shown to the domain experts at TOH to ensure they complied with the result of their experiments with different methods [9].
3. **Processes:** Add UCM-based processes to the model, which realize the goals by satisfying the tasks mentioned in the goal view. New goals are often discovered along the way, so goals and processes can be aligned.
4. **Scope:** Evaluate risks and select a subset of goals and processes for the application. The scope was set to the prospective surveillance solution and was supported by goal and process models, clinical experiments, and constraints of our team (i.e., very busy physicians and part-time development by a graduate student with little experience in the selected technologies). Having GRL for modeling the goals and UCM for the processes in separate layers makes solutions independent from

deployment structures and early commitment to architectures. This also increases the reusability of the model in different environments (hospitals, departments) and increases its flexibility and maintainability when requirements change.

5. **Implementation:** Use conventional software engineering and (Web-based) development methods to implement the application. The latter, in our context, uses a 3-tier architecture to increase the maintainability and usability of software assets.
6. **Pilot validation:** Use the application in a pilot study. Ours was tested by an observer nurse at TOH's Clinical Teaching Unit (CTU) for a month to collect data and solve usability and deployment issues. At the time of writing this paper, reviewer physicians have just started using the application for reviewing the data collected and extracting adverse event information, which has value to decision makers. The nurse is currently involved in a second pilot, this time for 3 months.

4 Patient Surveillance Application

This section provides several details on the process steps from the previous section as well as several observations and lessons learned.

4.1 Defining the Goals and Evaluating Strategies

We first considered the main stakeholders of the patient surveillance application, including The Ottawa Hospital (TOH), physicians, nurses, decision makers, patients, health care government agencies, and university researchers. As their goals are directly or indirectly influenced (sometimes in conflicting ways) by the use of this application, GRL diagrams were created (step 1 in section 3). Being the primary care provider, TOH was the most influential stakeholder in the definition of the scope of the application. Fig. 1 provides a very partial view of our GRL model¹ and shows some of TOH's high-level goals and their relations to government goals.

Having such a goal model at this stage helped us understand the expectations of different stakeholders and how they interact. A few counter-intuitive relations were also observed, e.g., that improving patient safety had a positive contribution on cost reduction because of overall decreased lawsuits and patient care costs (as explained in the connected GRL belief (ellipse) in Fig. 1, which acts as a comment). The model was also useful to understand the scope of the project and its risks. Improving patient safety is the high-level goal of TOH that is targeted by the surveillance application.

In a prospective surveillance method, improvement of patient safety is done by collecting data about adverse events, having it analyzed by knowledgeable reviewers (physicians), and making decisions on how to improve patient safety by decreasing the possibility of adverse event occurrence (e.g., by improving an existing health care business process). However, there are different ways of addressing each of these steps, and each has positive and negative contributions on the stakeholder goals. These were also modeled in GRL (not shown here) and GRL strategies were defined.

¹ Our current (and evolving) model is comprised of 87 GRL intentional elements and 10 GRL actors part of 12 GRL diagrams, of 72 UCM responsibilities and 14 UCM components part of 21 UCM diagrams, and of hundreds of relationships between these model elements. From our experience, this is an average-sized URN model.

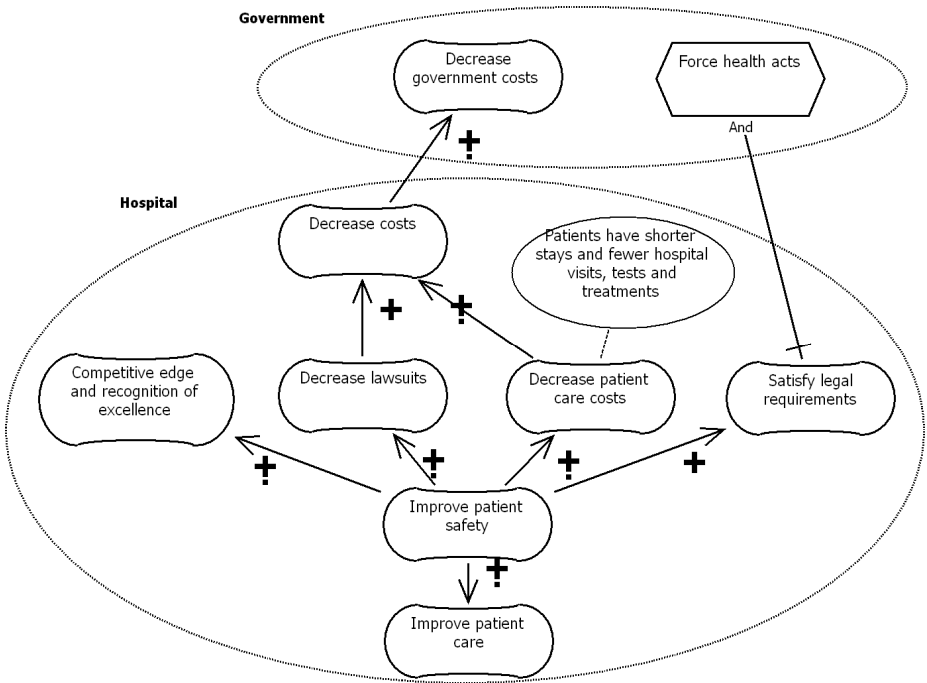


Fig. 1. Partial goal model from the TOH viewpoint

GRL models helped us reason about the requirements for patient safety. We have found that goal models are useful to communicate with stakeholders, especially domain experts, and discuss their requirements while conveying our own software engineering concerns. We used jUCMNav for comparing different alternatives by creating GRL strategies for each of them and then examining how they impact stakeholder objectives (step 2 in section 3). The visual evaluation feedback (GRL intentional elements become color-coded during an evaluation) helped stakeholders understand and assess such impact at a glance.

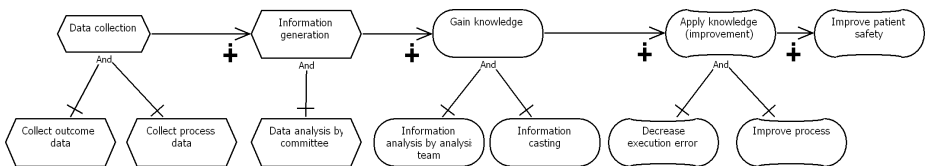


Fig. 2. Goals and high-level tasks of the prospective surveillance approach

Fig. 2 shows how the main goal of the TOH can be achieved through a set of soft-goals, goals and tasks. This sequence starts with data collection and ends with applying the knowledge of how to decrease the adverse events. The whole sequence will

result in improving patient safety. These sub-tasks and sub-goals are then considered, refined, assessed, and realized in more detail in the next development stages.

Having the goal model at such an early stage provided an opportunity to understand similarities between adverse event detection methods. Although we first focus on proactive surveillance (as this is the most cost-effective method) our application can be made flexible enough to support other and complementary methods at very little cost. For example, we recently received a request to consider voluntary incident reporting (where a physician reports a potential incident instead of an observer nurse) as an addition. This only affects the Data collection task of Fig. 2, and modifications to the model and its implementation can then be localized to small parts only.

4.2 Modeling the Process Satisfying the Goals

With UCM, we then model a process that satisfies the combination of intentional elements selected from the goal model (step 3 in section 3). This UCM view refines the goal model by sequencing tasks and providing additional details in a workflow-like representation that would be otherwise cumbersome to capture. This also paves the way towards architectural descriptions and the support of specific use cases.

Fig. 3 gives a high-level view of the overall adverse event management process.

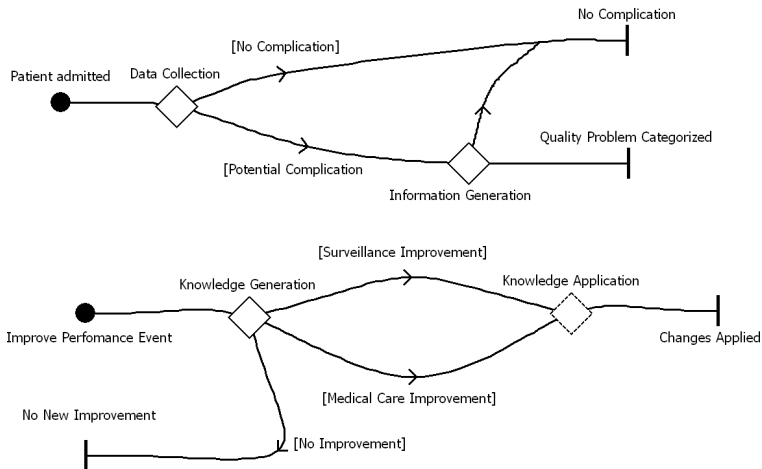


Fig. 3. High-level adverse event management process UCM

Stubs encapsulate the details of the sub-processes defining the four important intentional elements identified in Fig. 2, namely Data Collection, Information Generation, Knowledge Generation and Knowledge Application. This process view is independent of the underlying method of implementing each step. Also, UCM models offer the possibility to describe alternative process refinements with dynamic stubs (dashed diamond symbol). For instance, we have specified several possible ways of performing Knowledge Application, which are not shown here due to space constraints.

We have created UCM diagrams for all the stubs in Fig. 3. However, the implementation of the bottom half has been postponed to a second phase of the project

because of evolving requirements (we wanted to learn from the pilot study first) and the availability of development resources (step 4 in section 3).

As an example, Fig. 4 shows UCM diagrams detailing the Data Collection stub from Fig. 3, at three levels of abstraction. Part (a) is connected directly to the Data Collection stub and, given our focus on the prospective surveillance method, indicates that an observer nurse is involved. The Locate Health Care Quality Problem stub is refined in part (b), where the various responsibilities for observing processes and patient statuses are identified. As shown, many of them can be done in parallel or in any order. The three stubs in diagram (b) all contain the same diagram in part (c), which shows that sub-processes can be reused in many locations.

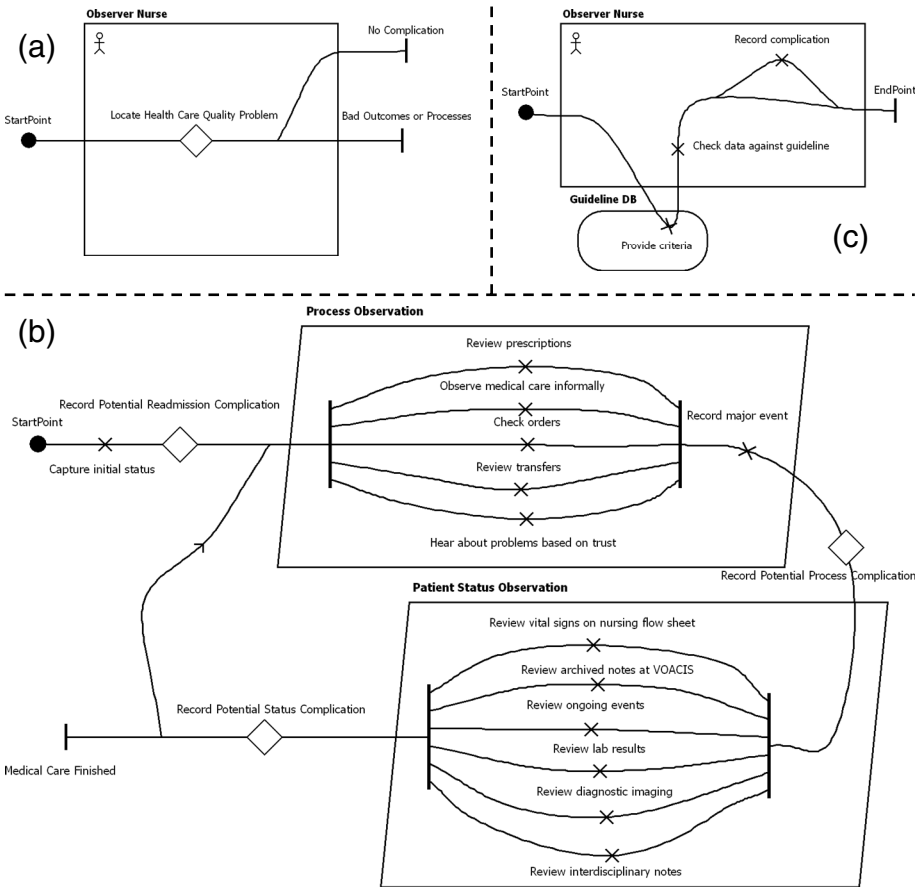


Fig. 4. UCM diagrams for prospective surveillance based data collection

Such models were useful when communicating with domain experts, but also with developers, which are more accustomed to use case models. Also, traceability from UCM elements to the GRL view helps them understand “why” the use cases are as

they are. UCM responsibilities can also be reassigned easily to other components along the way simply by dragging a responsibility and dropping it in the desired component box in a diagram, with jUCMNav. The cost of considering variations and of doing changes to the use case is then very low. Also, having different levels of details with sub-maps helps maintain the model when requirements change as it is possible to modify sub-models without breaking the general solution.

4.3 Software Architecture and Implementation

Considering the stakeholders' goals, requirements, and constraints, it was decided to use a Web application with a typical and loosely-coupled three-tier architecture to support the application (step 5 in section 3). This architecture is composed of a Web browser (on a tablet PC connected through a wireless network), a Web server containing a presentation layer (in ASPX) and a business logic layer, and a database server containing patient information and stored procedures. Different actors/roles (e.g., observer, reviewer, facilitator, and administrator) are given different tasks and access privileges. Constraints from TOH (who will eventually take over the maintenance of the application) included the use of the .NET framework and of MS SQL.

In our context, a Web application enabled the use of interfaces generally known to users, many of whom are very busy and require remote access, and eased application deployment. A central database enables the sharing of information across different users and across different steps of the business process.

We designed the database schema to support the goals and processes which were modeled in previous steps, also considering the types of requests users of future steps of our business process (the bottom part of Fig. 2) would likely perform. A UML class diagram was used to formalize the information about departments, patients, physicians, diseases and health problems, adverse events categories, observations, review decisions, etc. as well as many relationships such as who is in charge of a patient after admission.

To illustrate the interface, Fig. 5 shows the Web page that corresponds to the "Capture initial status" responsibility of the UCM in Fig. 4. An observer nurse uses this page to add patient data (some of which might eventually be obtained from existing operational system) and information about the care unit and physician to which the patient is assigned after admission. Many such pages were created for the various tasks and roles we identified.

4.4 Obstacles and Mitigations

Two major obstacles were encountered close to deployment time:

De-identification: A late requirement was added to satisfy health care privacy legislation and get permission to deploy the application for the pilot study. Identification information (e.g. name and patient identifier) needed to be stored behind the firewall of the hospital whereas the rest of the information needed to be on our database server in the research institute, behind a different firewall. The URN model was slightly evolved to reflect this requirement, and the code was changed to store the identifiers in a separate (XML) file instead of in the database. This file could then be located on

a different server, inside the hospital. Synthetic identifiers were generated and enabled the authorized users to access the information from both sources transparently and to present it in a combined way. However, this issue caused delays in the application deployment as well as stressful situations among stakeholders that could have been prevented by a more precise deployment plan.

Record Number:	<input type="text" value="TOH2701"/>		
First Name:	<input type="text" value="John"/>		
Last Name:	<input type="text" value="Smith"/>		
Date of Birth:	<input type="text" value="7"/>	<input type="text" value="May"/>	<input type="text" value="1965"/>
Brief history of presenting illness:	<input type="text" value="Accute pain"/>		
Gender	<input type="text" value="Male"/>		
Location prior to admission	<input type="text" value="ER"/>		
Admitting diagnosis	<input type="text" value="3 Degree Heart Block"/>		
Has the patient have any of the following chronic health problems?			
A recent MI in past 30 days	<input type="text" value="No"/>		
A prior MI	<input type="text" value="No"/>	DM	<input type="text" value="No"/>
CHF	<input type="text" value="No"/>	Hemiplegia	<input type="text" value="No"/>
CVD	<input type="text" value="No"/>	CRF	<input type="text" value="No"/>
PVD	<input type="text" value="No"/>	Cancer	<input type="text" value="No"/>
Dementia	<input type="text" value="No"/>	Metastatic cancer	<input type="text" value="No"/>
COPD	<input type="text" value="No"/>	Hematological malignancy	<input type="text" value="No"/>
		CTD	<input type="text" value="No"/>
		PUD	<input type="text" value="Yes"/>
		Cirrhosis	<input type="text" value="No"/>
		HIV	<input type="text" value="No"/>
		Psychiatric disorder	<input type="text" value="No"/>
Patient Responsibility Assignment			
Individual	<input type="text" value="Dr. Fariba"/> <input type="checkbox"/>		
CTU Admission Date:	<input type="text" value="7"/>	<input type="text" value="Jan"/>	<input type="text" value="2009"/>
CTU Admission Time:	<input type="text" value="15:00"/>		
	<input type="button" value="Cancel"/>	<input type="button" value="Save"/>	

Fig. 5. Snapshot of a data collection Web page

Downgrading the data layer: Initial requirements targeted the MS SQL Server 2005 database management system for the application’s data layer because the hospital planned to move from an earlier version (2000) to this one by deployment time. Our database was therefore created for version 2005, which is not backward compatible with version 2000. However, the upgrade to version 2005 was not available by deployment time. Our own server at the research center had version 2000, which was required by a companion Business Intelligence tool (Cognos 8). It was hence necessary to change some part of the data tier, and its separation from the business logic and presentation tier proved very useful at this stage as the remained untouched. Yet,

this introduced additional delays and stress as we spent time converting the data, tables and stored procedures to make them compatible with MS SQL 2000. This obstacle is an example of changes that can happen to all applications developed for industrial organizations by academic researchers. Continuous communication and visibility can decrease the risks related to unexpected changes to plans, and flexible architectures are essential to such collaborative projects.

5 Conclusion and Future Work

Joint projects and close collaboration between computer scientists and health care professionals are highly recommended by [11] in order to solve IT issues in this challenging area. This paper reported on an ongoing project targeting patient safety through the detection and analysis of adverse events, which can lead to the evolution of health care business processes. We have taken a goal-driven approach based on URN models that provided a suitable level of abstraction for productive university-industry collaboration in IT, where many stakeholders are busy and have very different background knowledge. Capturing and referring to “why” aspects helps to reduce the risks of misunderstandings, although this does not prevent all conventional obstacles (such as unexpected changes to deployment plans, see previous section) from happening. Such models are also resistant to change (because of their high level of abstraction) and flexible in case of changes (given their structure). They also provided design guidance for the later development steps, which led to a working application that takes advantage of e-technologies where paper-based approaches are often used. Other approaches based on goal models exist (such as van Lamsveerde’s [15]) but they seldom combine goals with more detailed processes or use cases as well as what is possible with URN, which for now also contains the only standard goal notation.

The results of the pilot study (step 6 in section 3) are very encouraging so far and the experiment has been renewed for a 3-month period. Few adjustments had to be made to the prototype, yet we plan to improve it on various quality aspects such as usability, robustness, security and scalability. We also plan to deploy it in other hospital departments and even in a different hospital, where the culture, regulations and business processes are different. We expect the URN models to be quite reusable (given their generality) and the application itself should easily adaptable since we have made it customizable (in terms of departments, types of diseases, types of adverse events, etc.) from the beginning. We also plan to evolve the URN model to take advantage of KPI extensions proposed in [10] and use it for performance modeling.

Acknowledgments

This work was supported by a Collaborative Health Research Project grant from CIHR and NSERC (Canada) on *Performance Management at the Point of Care: Secure Data Delivery to Drive Clinical Decision Making Processes for Hospital Quality Control*. We are thankful to the OHRI personnel and to the following students for their help in developing and deploying the surveillance application: J. Blais, R. D’Angelo, M. Garzon, and R. Bougueng Tchemeube.

References

1. Amyot, D.: Introduction to the User Requirements Notation: Learning by Example. *Computer Networks* 42(3), 285–301 (2003)
2. Baker, G.R., Norton, P.G., Flintoft, V., Blais, R., Brown, A., Cox, J., et al.: The Canadian Adverse Events Study: the incidence of adverse events among hospital patients in Canada. *CMAJ* 170(11), 1678–1686 (2004)
3. Cullen, D.J., Bates, D.W., Small, S.D., Cooper, J.B., Nemeskal, A.R., Leape, L.L.: The incident reporting system does not detect adverse drug events: a problem for quality improvement. *Joint Commission Journal on Quality Improvement* 21(10), 541–548 (1995)
4. Forster, A.J., Halil, R.B., Tierney, M.G.: Pharmacist surveillance of adverse drug events. *American Journal of Health-System Pharmacy* 61(14), 1466–1472 (2004)
5. Gandhi, T.K., Weingart, S.N., Borus, J., Seger, A.C., Peterson, J., Burdick, E., et al.: Adverse Drug Events in Ambulatory Care. *The New England Journal of Medicine* 348(16), 1556 (2003)
6. Gurwitz, J.H., Field, T.S., Avorn, J., McCormick, D., Jain, S., Eckler, M., et al.: Incidence and preventability of adverse drug events in nursing homes. *American Journal of Medicine* 109(2), 87–94 (2000)
7. ITU-T – International Telecommunications Union: Recommendation Z.151 (11/08): User Requirements Notation (URN) – Language definition. Geneva, Switzerland (2008)
8. Liu, L., Yu, E.: Designing Information Systems in Social Context: A Goal and Scenario Modelling Approach. *Information Systems* 29(2), 187–203 (2004)
9. Michel, P., Quenon, J.L., de Sarasqueta, A.M., Scemama, O.: Comparison of three methods for estimating rates of adverse events and rates of preventable adverse events in acute care hospitals. *British Medical Journal* 328(7433), 199–203 (2004)
10. Pourshahid, A., Chen, P., Amyot, D., Forster, A.J., Ghanavati, S., Peyton, L., Weiss, M.: Toward an integrated User Requirements Notation framework and tool for Business Process Management. In: 3rd Int. MCeTech Conference on eTechnologies, Montréal, Canada, January 3–15. IEEE Computer Society, Los Alamitos (2008)
11. Stead, W.W., Lin, H.S.: Computational Technology for Effective Health Care: Immediate Steps and Strategic Directions. In: Committee on Engaging the Computer Science Research Community in Health Care Informatics, National Research Council, USA. National Academies Press, Washington (2009)
12. The Institute of Medicine. To err is human: building a safer health system. National Academy Press, Washington D.C (2000)
13. Univ. of Ottawa: jUCMNav 3.2 (2008), <http://jucmnav.softwareengineering.ca/jucmnav/>
14. Weiss, M., Amyot, D.: Business Process Modeling with URN. *International Journal of E-Business Research* 1(3), 63–90 (2005)
15. van Lamsweerde, A.: Requirements engineering: From System Goals to UML Models to Software Specifications. John Wiley & Sons, Chichester (2009)

Global Location-Based Access to Web Applications Using Atom-Based Automatic Update

Kulwinder Singh and Dong-Won Park

Department of Information and Communications Engineering, PaiChai University
439-6, Doma-Dong, Seo-Gu, Daejeon, South Korea
{singh,dwpark}@pcu.ac.kr

Abstract. We propose an architecture which enables people to enquire about information available in directory services by voice using regular phones. We implement a Virtual User Agent (VUA) which mediates between the human user and a business directory service. The system enables the user to search for the nearest clinic, gas station by price, motel by price, food / coffee, banks/ATM etc. and fix an appointment, or automatically establish a call between the user and the business party if the user prefers. The user also has an option to receive appointment confirmation by phone, SMS, or e-mail. The VUA is accessible by a toll free DID (Direct Inward Dialing) number using a phone by anyone, anywhere, anytime. We use the Euclidean formula for distance measurement. Since, shorter geodesic distances (on the Earth's surface) correspond to shorter Euclidean distances (measured by a straight line through the Earth). Our proposed architecture uses Atom XML syndication format protocol for data integration, VoiceXML for creating the voice user interface (VUI) and CCXML for controlling the call components. We also provide an efficient algorithm for parsing Atom feeds which provide data to the system. Moreover, we describe a cost-effective way for providing global access to the VUA based on Asterisk (an open source IP-PBX). We also provide some information on how our system can be integrated with GPS for locating the user coordinates and therefore efficiently and spontaneously enhancing the system response. Additionally, the system has a mechanism for validating the phone numbers in its database, and it updates the number and other information such as daily price of gas, motel etc. automatically using an Atom-based feed. Currently, the commercial directory services (Example 411) do not have facilities to update the listing in the database automatically, so that why callers most of the times get out-of-date phone numbers or other information. Our system can be integrated very easily with an existing web infrastructure, thereby making the wealth of Web information easily available to the user by phone. This kind of system can be deployed as an extension to 911 and 411 services to share the workload with human operators. This paper presents all the underlying principles, architecture, features, and an example of the real world deployment of our proposed system. The source code and documentations are available for commercial productions.

Keywords: VoiceXML, CCXML, AtomPub, Asterisk Server, IP-PBX, GPS, Interactive Directory Service, Atom Feed.

1 Introduction

Users prefer online business directories, which can save lot of time if the listings are reliable and are regularly or automatically updated. In today's world online directories are preferred, but the directory services do not have a provision to update the information automatically. For example, if a user wants to find the nearest and cheapest gas station from a directory service using a phone, then the directory service must have a mechanism for updating the listings hourly, because the gas price changes frequently. In this context, the directory system typically access information from several remote databases. Accessing information from several databases is even more difficult than retrieving information from a single local database. The databases could be different such as Mysql or Oracle, use different schema, and employ different communication protocols. If the databases are located within different companies, the problems are even more severe. Most enterprises are not willing to permit other enterprises to access their databases directly. Moreover, the database communications between the different datacenters require penetrating firewalls. In order to alleviate the above mentioned problem, we propose a light weight data integration using Atom [2] protocol, but RSS (Really Simple Syndication) could be deployed instead. Our current data integration infrastructure retrieves the information from different database technologies.

Our paperwork contributes as follows:

1. Implementation of Atom protocol for updating the listings in the database of the directory system automatically.
2. Implementation of a Widget for retrieving the location (GPS coordinates) of a mobile user and Caller ID over the HTTP (bypassing the Telco phone network).
3. The entire infrastructure utilizes open source technologies. So, it is really inexpensive to build such a system.
4. A mechanism for validating the phone numbers and addresses automatically of all business listings in the directory's database.

Currently, directory services [6] do not have a mechanism for updating and validating the listing information automatically such as phone numbers, addresses. In addition, the existing GPS enabled services are Telco's dependent and expensive to build because most of the services use proprietary resources. In contrast, our GPS enabled application is totally independent of Telco's monopoly and it is very cost effective to deploy with an existing system.

2 The System Architecture of the VUA

The graphical user interfacing GUI uses HTML tags that are rendered into a visible page by a Web browser. Similarly, a voice application consists of XML (VoiceXML, CCXML) tags which become an interactive voice application when processed by the VUA. All you need to do is write the application as we have mentioned in our previous publication [1], map it to a phone number in the VUA Application Manager, and

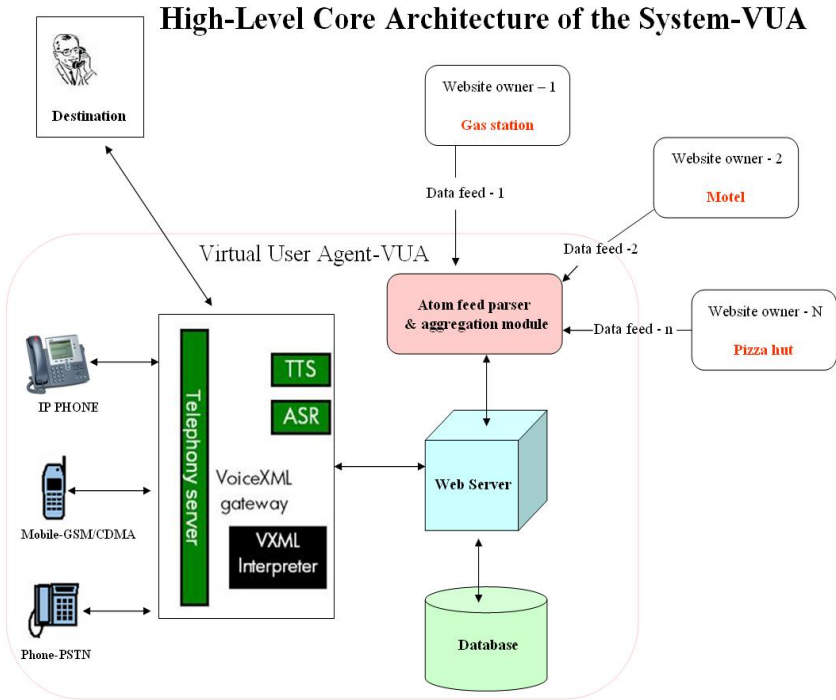


Fig. 1. The core architecture of the VUA

Architecture of the Atom Feed Distribution for Directory Services

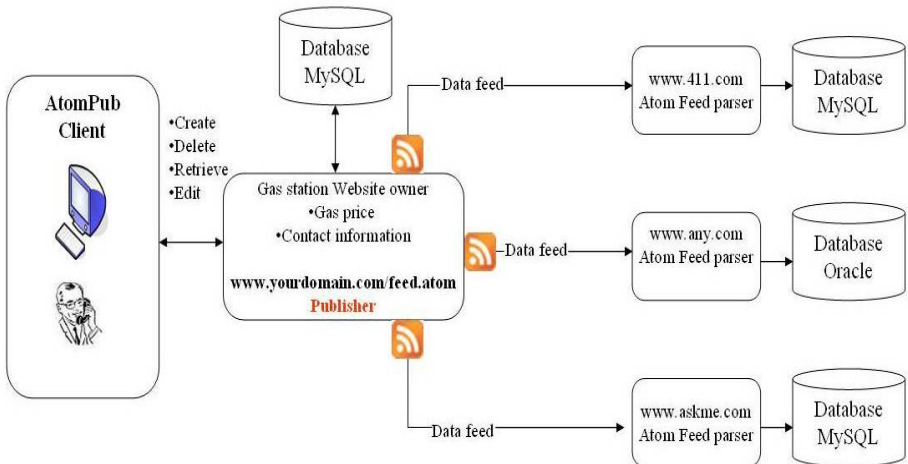


Fig. 2. The Atom feed distribution for various directories services

call the application. Fig. 1 depicts the architecture of our proposed system. The application can be called from any phone [1] as shown in Fig. 1. Once the call is placed to the VUA, the VUA answers the call and sends an HTTP request to a web server to fetch the voice application's contents. A web server returns the request in the form of a VoiceXML or CCXML document. Voice gateway [1] interprets the VoiceXML document, and connects the caller's destination number automatically. The aggregation module and Atom feed parser take as input Atom feeds from multiple data sources as shown in Fig. 1 and 2, which simulate incoming data from the AtomPub client. The Atom feed parser, and aggregation module is written in PHP.

We consider all the data feeds are homogeneous because we use a standard Atom feed structure. After collecting the Atom feeds, the integration module integrates the data supplied by the feeds into a schema that matches the target database's schema, as shown in Fig. 1 and 2. In order to connect a data source with various directories' databases as shown in Fig. 2 a standard Atom feed can be used. It means a website owner can create an Atom feed and can send the data to various directories' services at once. This process will update the target database if any change occurs in the source database.

3 Location-Based Call Routing

Users often want to find the nearest Pizza Hut, restaurants, gas station by price, clinic, motel by price, Wi-Fi hotspots, food/coffee and banks/ATMs etc. We propose a technically feasible idea that can be implemented for achieving the aforementioned goal. First we change the Geodetic Coordinates to the Cartesian coordinates using the following equations.

$$\begin{aligned}
 x &= (v+h)\cos f \cos l \\
 y &= (v+h)\cos f \sin l \\
 z &= \{(1-e^2)v+h\}\sin f
 \end{aligned}
 \qquad
 \text{Where }
 \begin{aligned}
 v &= a \sqrt{1 - e^2 \sin^2 f}^{0.5} \\
 a &= 6378137 \text{ m} \\
 f &= 1 / 298 .25722563 \\
 e^2 &= 2 f - f^2 \\
 h &= \text{height above sea level (assumed to be 100 m)} \\
 f &= \text{latitude (radians)} \\
 l &= \text{longitude (radians)}
 \end{aligned}$$

We take the origin to be the center of the Earth, the z-axis through the poles and the x-axis at longitude 0. Let R be the radius of the Earth. If a point has latitude phi (positive for North, negative for South) and longitude theta (positive for East longitude, negative for West), we convert to rectangular coordinates (x, y, z). If theta=0, then by basic trigonometry, x=R cos phi, y=0, z=R sin phi. When we vary theta (phi fixed), and we sweep out a circle with fixed z, so trigonometry gives x=R cos phi cos theta, y=R cos phi sin theta, z=R sin phi (the general formula).

Now if the distances are small enough that we can ignore the curvature of the earth (as you see in Fig.3 we consider a single zip code or less than a half kilometer). So, we can use the Euclidean distance formula as follows.

In **Euclidean** three-space, the distance d_1 between points (X_1, Y_1, Z_1) and (X_i, Y_i, Z_i)

$$d_1 = \sqrt{(X_1 - X_i)^2 + (Y_1 - Y_i)^2 + (Z_1 - Z_i)^2}$$

$$d_n = \sqrt{(X_n - X_i)^2 + (Y_n - Y_i)^2 + (Z_n - Z_i)^2}$$

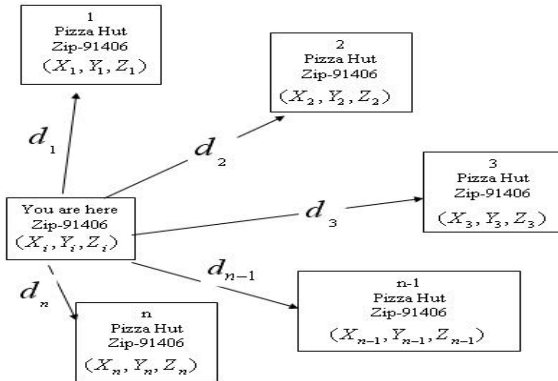


Fig. 3. Air distance measurements for call routing

As the distance being measured is minimal as you see in Fig. 3. Moreover, we are not going to provide the driving directions. In addition, we also assume that if an air distance is smaller, then the location would be closer. However, there are certain exceptions such as a traffic jam, a river between the caller and the destination. So, we believe that it is unnecessary to calculate the actual distance & time by road in our case. Actually, we just want to know, which location is the nearest with respect to the caller based upon the air distance, because we want to route the call to Pizza Hut for placing an order, then it is always enough to use the Euclidean formula, since shorter geodesic distances (on the Earth's surface) correspond to shorter Euclidean distances (measured by a straight line through the Earth). In that case, the square root is not required, since the squaring function is monotonic.

On the other hand, if we consider a large area (500 miles), we would compute the actual distance & time by road, and we can use the formula from vector calculus that is the cosine of the angle gamma between the vectors is their dot product /R^2. That is, delete the R from the above formulas to get unit vectors, then $\cos \gamma = x_1.x_2 + y_1.y_2 + z_1.z_2$. The geodesic distance between the points is then $R \gamma$ (where we measure the angle in radians).

3.1 Implementation and Use of the Mobile Web Widget

The availability of inexpensive, small, low-power GPS chips and reliable network has moved the industry into a growth phase for developing location-based applications such as vehicle tracking, child tracking navigation and route finding. Mobile users are interested in buying location-aware applications. The location-based application

Retrieving Longitude and Latitude from the Mobile device

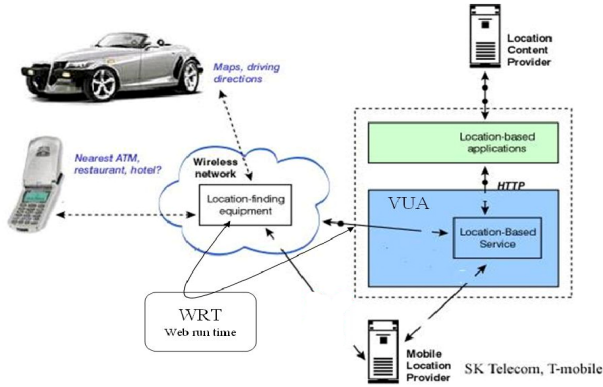


Fig. 4. Use of WRT widget

provider needs to go through Telco network as shown in Fig. 4 in order to execute the applications. Now a small or mid-sized provider might have trouble paying a large sum of money for Telco’s dependent application. We introduce an approach for retrieving longitude and latitude directly from the mobile device without interacting with telecom as you see in Fig. 4 and 5. Nokia handheld devices support few positioning methods such as Bluetooth GPS, assisted GPS, integrated GPS and network-based GPS. We use a Web Runtime (WRT) widget over the WIFI/GPRS network as shown in Fig. 4, and we develop the widget using Nokia SDK v.0.9 S60 5TH edition. We have tested the widget on Nokia 5800 successfully using Google’s geodetic data. The widget only supports Nokia. WRT widgets are lightweight mobile applications developed using standards-based Web technologies, such as XHTML, CSS, JavaScript, and Ajax. Let’s consider a scenario whereby a mobile caller wants to find the nearest clinic. The caller needs to enter his or her phone number and hit the “call me” button you can see in Fig. 5. Our VUA receives the request with geodetic data and caller ID over the http, and returns the call to user immediately. The user can communicate with the VUA, and the VUA establishes the call between the nearest clinic and the user.

Table 1. WRT Supported Properties

Name	Type	Use	Description
DisplayName	String	Required	Specify the actual name of the widget listed on the Installed application bar
Identifier	String	Required	Specify unique string identifier for the widget in reverse domain format
MainHTML	String	Required	Specify the name of the main HTML page that points to the widget
AllowNetworkAccess	Boolean	Optional	Specify access to the network based resources from the widget
ShortVersionString	String	Optional	Specify release version of the widget bundle
Version	Number	Optional	Specify build version of the widget bundle

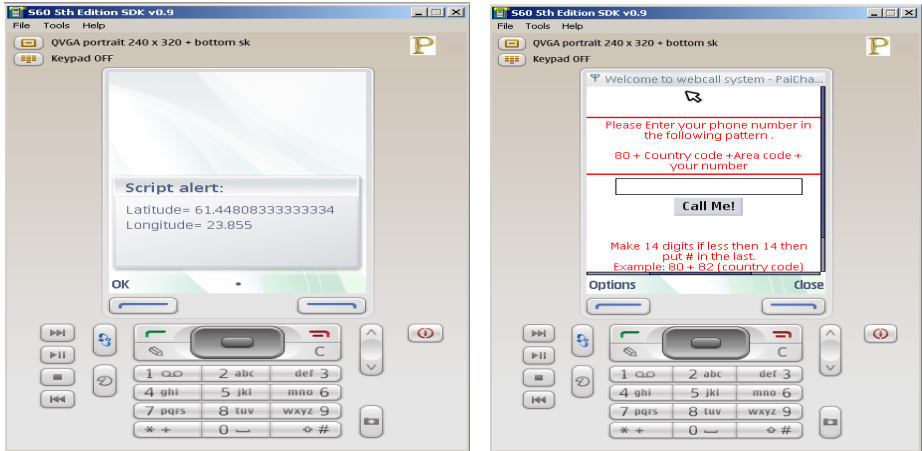


Fig. 5. Retrieving Longitude and Latitude from the Mobile device

3.2 Widget Component Structure

A widget is constructed by bundle of files.

- info.plist (mandatory)
- icon.png
- [name].html (mandatory)
- [name].css
- [name].js

A widget project is a file-system directory. In which widget’s component files are stored. Widget’s mandatory files and the icon.png (if included) MUST be located at the root directory of a widget project.

3.3 Info.plist-Property of a Widget

A manifest file in XML format contains the property and configuration information of a widget as shown in the following example. (see table 1 also)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Nokia//DTD PLIST 1.0//EN"
"http://www.nokia.com/NOKIA_COM_1/DTDs/plist-1.0.dtd">
<plist version="1.0">
<dict>
<key>DisplayName</key>
<string>WidgetName</string>
<key>Identifier</key>
<string>com.company.widget.project</string>
<key>MainHTML</key>
<string>Main.html</string>
</dict>
</plist>
```

3.4 Icon.png

A PNG image is used as an application icon in the mobile device. In order to execute the application user can simply click on the application icon after the installation. The recommended size of the icon is 88x88 pixels. If icon is missing and then the default S60 application icon is used as you see in Fig. 6.

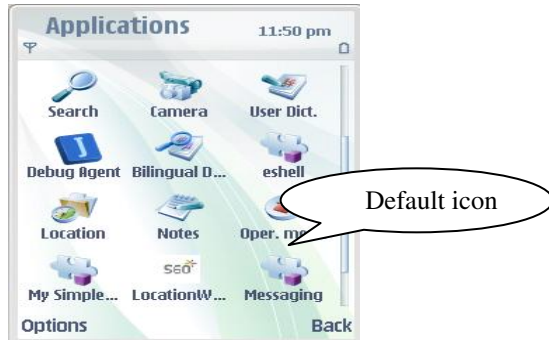


Fig. 6. Installed applications

4 Atom

Atom [2, 4, 7] is an XML-based document format that describes lists of related information known as "feeds". Feeds are composed of a number of items, known as "entries", each with an extensible set of attached metadata. For example, each entry has a title. It was originally developed based on RFC 4287 for web logs as an open alternative to "frozen" RSS 2.0 format. Now it has been used in web developments. Atom provides a concrete implementation and can be further sub classed for particular cases. In other words, we say that it is extensible through XML namespaces. We choose Atom for light weight data integration, because it provides a better platform than existing standards for exchanging data. There is another technology called SOAP, but it is not user friendly, requires a special chain of tools and does not have a very simple format. On the other hand, Atom has a very simple format, does not require any special tools and is very user friendly.

According to our prototype system, we need the following standards for making the system universally acceptable:

- A standard identification system
- A standard interaction protocol
- Standard data exchange formats
- A Standard way of handling states

Atom provides these standards. According to aforementioned standards, a user, who has a listing or wants to publish the information in the directory service, needs to create an Atom feed as shown in Fig. 2. The Atom feed parser of the directory service collects the feed, checks it periodically, and automatically updates the information in

the target database if any change occurs in the source database. Our proposed idea saves a lot user's time because the user does not need to edit the listing in all the directory services. Moreover, the caller gets up-to-date information from the directory services. The current commercial directory services do not have this feature. A simple data model of Atom is show in Fig. 7

A Simple Data Model-ATOM

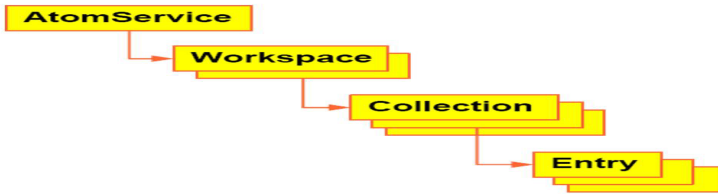


Fig. 7. A simple data model of Atom

- Entry represents an individual piece of content (post, article, page, image, video, document, etc.). Moreover Entries can be related to other entries or Feeds.

This resource represents a specific resource

(i.e. <http://example.org/posts/1234>)

Content type:

`application/atom+xml;type=entry`

- Feed represents a collection of Atom entries, can be related to other feeds or entries.

This resource represents a specific

collection of resources (i.e. <http://example.org/posts>).

Atom collections are feeds

Content type:

`application/atom+xml;type=feed`

- Categories provide metadata to describe an Atom entry; Atom itself doesn't define usage of categories. Category documents describe the categories that are allowed in collections.

Content type:

`application/atomcat+xml`

- A service discovery document that describes the locations and capabilities of Collections. Usually the entry point of the Atom web Service

Content type:

`application/atomsvc+xml`

Atom does not provide or recommend an auth mechanism, but authentication info through HTTP headers can be sent on each request requiring authenticated access. Atom does not provide any encryption, but SSL can be for all sensitive communication. Atom Entry and Feed Documents can contain XML Digital Signatures. Moreover, data can be encrypted using XML encryption.

Atom feed loader is available at <http://biometrics.pcu.ac.kr/atom/atom-links.php> (collection of Atom Uri's). The example of the atom feed document is available at <http://biometrics.pcu.ac.kr/atom/atom.xml>

```

<?xml version="1.0" encoding="UTF-8" ?>
<feed xmlns="http://www.w3.org/2005/Atom"
      xmlns:sy="http://purl.org/rss/1.0/modules/syndication/"
      xml:lang="en">
  <title>Kulwinder Singh: Homepage</title>
  <link rel="alternate" type="text/html"
href="http://w2.pcu.ac.kr/~singh"/>
  <link rel="self" type="application/atom+xml"
href="http://biometrics.pcu.ac.kr/atom/atom.xml"/>

  <id>http://w2.pcu.ac.kr/~singh/index.php</id>
  <updated>2007-02-10T16:15:50+01:00</updated>

  <entry xml:lang="en">
    <title>Contact information</title>
    <link rel="alternate" type="text/html"
href="http://w2.pcu.ac.kr/~singh/contact.html" />
    <updated>2008-02-10T16:15:50+01:00</updated>
    <id>http://w2.pcu.ac.kr/~singh/contact.html</id>
    <author><name>Webmaster</name></author>
    <category term="contact" label="contact"/>
    <summary>Phone number: 253-242-2448</summary>
  </entry>
</feed>

```

4.1 AtomPub Protocol (APP)

The Atom Publishing Protocol [7] is an application-level protocol for publishing and editing Web Resources using HTTP and XML. APP uses GET, POST, PUT and DELETE as show in Fig. 2, and uses HTTP response codes for error handling.

For example:

- 200 OK
- 404 Not found
- 403 Forbidden

It has built in support for arbitrary media types – locations, pictures, pod casts, etc., and HTTP caching mechanisms. Moreover, it fully utilizes web infrastructure.

5 Results

5.1 Evaluation of the Atom Feed

We evaluate the performance of our light weight data integration module as follows:

We asked 15 users to create the atom feed for listing in our directory database and Atom parser collected the feed and checked it periodically. Then we asked 5 users to update the listing information using AtomPub [7] as show in Fig. 2. We found that the directory database was successfully updated. It means that any change in the source database reflects in the target database. If commercial directory services could use this approach, then the caller would always get up-to-date information.

5.2 Evaluation of the VOIP Providers With Asterisk

There are many choices when you want to select a VOIP provider to terminate your Asterisk PBX SIP or IAX2. One of the most important factors in this selection is the network latency between your PBX and the VOIP provider. The route between you

and your VOIP provider is one of the most important considerations when selecting a provider. In this section we describe a simple method for evaluating this connection using Asterisk.

Asterisk [3] includes a feature that can be used to monitor the latency between your system and a peer or friend. This feature is enabled by setting the qualify setting in the `iax.conf` and `sip.conf` configuration file. Valid options are `yes`, `no`, or a time in milliseconds. If qualify is enabled, NOTIFY messages are sent periodically to the peer and the latency between replies is measured. The peer is determined unreachable if the number of milliseconds is greater than the qualify value or 2,000 if qualify is set to `yes`. If a peer is unreachable, events are logged in `/var/log/asterisk/messages` as shown below: (see Fig. 8,11 also)

```
[Dec 24 07:43:30] NOTICE[3346] chan_sip.c: Peer 'callcentric'
is now UNREACHABLE! Last qualify: 219
[Dec 24 07:43:54] NOTICE[3346] chan_sip.c: Peer 'callcentric'
is now Reachable. (304ms / 2000ms)
[Dec 24 06:54:25] NOTICE[3346] chan_sip.c: Peer 'pennytell'
is now Lagged. (3159ms / 2000ms)
[Dec 24 06:54:35] NOTICE[3346] chan_sip.c: Peer 'pennytell'
is now Reachable. (160ms / 2000ms)
[Dec 23 05:01:20] NOTICE[3346] chan_sip.c: Peer 'smsdis' is
now Lagged. (3558ms / 2000ms)
[Dec 23 05:01:30] NOTICE[3346] chan_sip.c: Peer 'smsdis' is
now Reachable. (314ms / 2000ms)
```

We are currently using many different VOIP providers and we get qualify notice messages about once every day or two with the qualify set to 360ms. This is very useful information as it allows us to determine which provider has the most reliable route from our IP-PBX. If you get very few qualify NOTICE messages, you can assume your VOIP connection will be fairly reliable.

Having connections with different providers also tells us when the congestion is with our internet connection or if it is something beyond our connection. If all the providers go out at the same time, it is most likely that our internet connection has some problems. We have never seen this scenario, so we can conclude that our internet connection is fairly reliable and the congestion problems are most likely between us and the VOIP provider or problems with the VOIP provider.

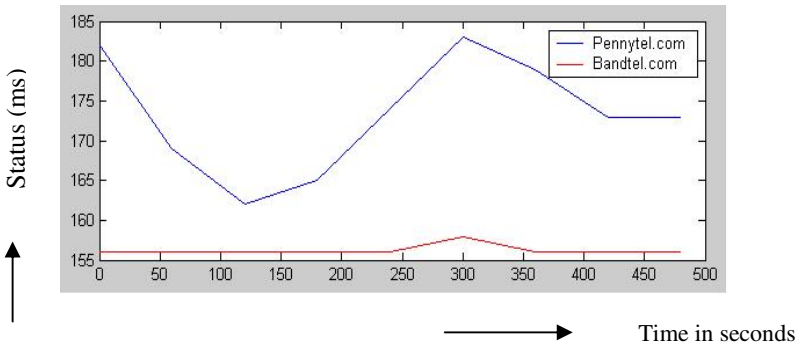


Fig. 8. evaluation of VOIP provider

Name/username	Host	Dyn	Nat	ACL	Port	Status
VoIPtalk_me/84484270	77.240.48.94		N		5060	Unmonitored
t-pad-me/1545970	213.40.29.3		N		5060	OK (296 ms)
smedis/dwpark	194.120.0.198		N		5060	OK (312 ms)
gipbroker/5633	64.34.162.221		N		5060	Unmonitored
pennytell/8889182974	202.85.243.87		N		5060	OK (160 ms)
orbtalk/7217654	217.14.132.178		N		5060	OK (318 ms)
orb4-kamal/7226997	217.14.132.178		N		5060	OK (318 ms)
orb3/7218831	217.14.132.178		N		5060	OK (316 ms)
orb2/7218649	217.14.132.178		N		5060	OK (317 ms)
orb1/7217658	217.14.132.178		N		5060	OK (316 ms)
orb-notti/7229242	217.14.132.178		N		5060	OK (317 ms)
orb-call/7227502	217.14.132.178		N		5060	OK (317 ms)
netplan/testaccount	62.169.138.130		N		5060	OK (296 ms)
messagenet/5338880	212.97.59.76		N		5061	OK (307 ms)

Fig. 9. Asterisk CLI – peer status

6 Conclusion

We conclude that the call quality may differ depending upon the different feature sets (e.g., codecs) and network bandwidth available. In order to get a stable connection with a VoiceXML gateway, the call should pass through minimum VOIP gateways. We also conclude as follows:

- Lightweight architecture for data integration
- User friendly, cost effective and simple
- Interoperability due to Atom/XML
- Improved reliability due to update information in the database

The directory service provider can route the toll free calls thru ENUM [1] in order to save money, and earn revenue by selling advertise time that takes the place of a ring tone while the callers waits for their call to be connected.

References

1. Singh, K., Park, D.-W.: Economical Global Access to a VoiceXML Gateway Using Open Source Technologies (Coling 2008). In: Proceedings of the workshop on Speech Processing for Safety Critical Translation and Pervasive Applications, Manchester (UK), August 2008, pp. 17–23 (2008)
2. Nottingham, M., Sayre, R.: The Atom syndication format. RFC 4287, <http://www.ietf.org/rfc/rfc4287.txt>
3. Meggelen, J.V., Madsen, L., Smith, J.: Asterisk: The Future of Telephony, 2nd edn. O'Reilly, Sebastopol (2007)
4. <http://www.atomenabled.org/>
5. Ruiz, Q., Sanchez, M.: Design of a VoiceXML Gateway. In: Fourth Mexican International Conference on Computer Science, p. 49 (2003)
6. Lehtinen, G., Safra, S., et al.: IDAS: Interactive Directory Assistance Services. In: Proceedings of the COST249 ISCA Workshop on Voice Operated Telecom Services, Gent, Belgium, May 2000, pp. 51–54 (2000)
7. Gregorio, J. (ed.): RFC 5023 (2007), <http://www.ietf.org/rfc/rfc5023.txt>

Toward a Framework for Dynamic Service Binding in E-Procurement

Maryam Ashoori, Benjamin Eze, Morad Benyoucef, and Liam Peyton

University of Ottawa
550 Cumberland St. Ottawa, Ontario, Canada.
(m.ashoori, beze, benyoucef, lpeyton)@uottawa.ca

Abstract. In an online environment, an E-Procurement process should be able to react and adapt in near real-time to changes in suppliers, requirements, and regulations. WS-BPEL is an emerging standard for process automation, but is oriented towards design-time binding of services. This missing issue can be resolved through designing an extension to WS-BPEL to support automation of flexible e-Procurement processes. Our proposed framework will support dynamic acquisition of procurement services from different suppliers dealing with changing procurement requirements. The proposed framework is illustrated by applying it to health care where different health insurance providers could be involved to procure the medication for patients.

Keywords: E-Procurement, Service Oriented Architecture Web service, BPEL, Health care.

1 Introduction

One of the major challenges of procurement is to react quickly to the needs of the organization with less people involved at a reduced processing cost. To address this challenge, researchers have looked to leverage the power of web technologies to automate procurement processes with existing trading partners. It leads to a trend for organizations of all sizes to migrate onto the Internet with resulting changes to procurement business processes [1, 2]. E-Procurement is defined by [3] as a comprehensive process in which organizations either establish agreements for the acquisition of products/services (contracting) or purchase products/services in exchange for payment (purchasing). Service-oriented architecture (SOA) [4] is emerging as the premier integration and architecture framework for Business to Business (B2B) collaboration in today's complex and heterogeneous computing environments. The inherent complexity of e-Procurement that involves from interaction across multiple parties with heterogeneous implementation technologies can be effectively handled by SOA [5]. SAP Discovery System [6] is a good sample implementation of a service-enabled procurement scenario. Web services are the preferred standards-based way to realize SOA [7]. One of the key benefits of web services is interoperability, which allows different distributed web services to run on heterogeneous platforms. This interoperability is gained through a set of XML-based open standards, such as WSDL, SOAP,

and UDDI [8]. These standards provide a common approach for defining, publishing, and using web services. Chen et al [9] demonstrates how deploying web services for procurement reduces integration costs.

BPEL(Business Process Execution Language) [10] introduced by OASIS provides a standards-based XML language for scripting the steps in a procurement process as an orchestration of interactions with web services.

BPEL is appropriate for highly targeted or constrained systems as it does not make any assumptions about the WSDL binding. It statically binds specific web services known at design time. Procurement applications are, however, inherently dynamic as business situations between the parties change over time or there are new parties that a BPEL defined e-Procurement process should be able to negotiate with. However, BPEL cannot flexibly configure and re-configure the e-Procurement on the fly. Namely, for any given service in a BPEL defined e-Procurement process, there may be many different competing vendors, and resources that can fill that "role". For example, in health care medicine procurement, different health insurance providers can be the ones paying for it, and they can each have different rules about what is allowed and what the process is for providing payment. Sometimes the procedure is totally covered, sometimes half, sometimes not at all (so the patient pays the difference, or the spouse's health insurance plan kicks in). It can all change the e-Procurement business process. Consequently, fully automatic, dynamic binding of services is a prerequisite for the formation of such dynamic e-Procurement processes.

In this paper, we present a framework for dynamic binding of e-Procurement services. The rest of this paper is structured as follows. Section 2 will review the evolution of technology in the context of SOA based e-Procurement while the architecture and design of the proposed framework will be discussed in Section e. Section 4 will discuss the criteria for evaluating B2B interactions in e-Procurement with respect to the proposed framework. This will highlight the contributions of our work followed by conclusions and a discussion of remaining challenges and future work.

2 Related Standards and Technologies

The earliest solutions for B2B interactions are point-to-point Electronic Data Interchange (EDI) [11] and component-based B2B solutions (CORBA[12], DCOM[13], and EJB[14]), followed by inter-enterprise workflows[14] and XML-based B2B interaction frameworks (eCO [15], BizTalk [16]), cXML[17], RosettaNet[18], and ebXML[19]). The EDI model as a data-oriented standard to describe business documents has been the predominant model for procurement solutions. However, existing implementations demonstrate that interaction based upon data alone is not enough to automate business communication[20]. EDI and XML are both based on a data interchange approach. However, web services [9] take a service- and process-oriented approach in addressing integration issues in procurement. Existing XML-based frameworks for B2B interactions mostly deal with a content layer to standardize the message format for industries (e.g., eCO, cXML) and business process layers to standardize the set of common business process specifications that are shared by multiple industries (e.g., RosettaNet, ebXML). These frameworks sometimes overlap or even compete with each other [14].

ebXML is one of the most ambitious business process standards to provide a flexible, open infrastructure for electronic business. However, there are still some interoperability problems with ebXML as ebXML suffers from a lack of a common XML library or framework to enable trading partners to unambiguously identify and exchange business documents in specific contexts [21]. Web services improve existing XML-based frameworks by standardizing the protocol and explicitly representing the message format using WSDL and making it publicly accessible and discoverable by UDDI [22]. Nevertheless, web services will only have advantages over existing technologies if service binding can be performed dynamically.

WSFL (Web Services Flow Language)[23], proposed by IBM, is an XML-based language that describes web services compositions as a business process. IBM WSFL was later replaced by BPEL4WS (Business Process Execution Language for Web Services) as a standard language for defining the methods of XML messaging, operating XML data structures, and handling events and exceptions. Later, OASIS introduced WS-BPEL [10] as the standard interoperable integration model to facilitate the expansion of automated process integration in B2B collaborations. BPEL is appropriate for highly targeted or constrained systems as it does not make any assumptions about the WSDL binding.

The fully automatic, dynamic binding of services is a prerequisite for the formation of flexible BPEL defined e-Procurement services. However, few literature sources discuss how to design and implement flexible procurement applications with dynamic binding to business processes. Klein and Konig-Ries [24] believes that the most promising approach for a completely automated service binding is a semantic service description. Current technologies for expressing requests in these languages (e.g. XSRL [25], DQL[26]) are not designed for the task of automatic service trading. Klein has argued that a specialized service request language including preferences is the major prerequisite for automatic service binding. With such a language it becomes possible to describe service requests precisely yet flexibly. Although it covers comprehensively the service request part, it still does not address mapping the discovered services to the BPEL defined business process. Oracle [27], Active Endpoints [28], and IBM [29] have addressed dynamic service binding in BPEL by utilizing endpoint references[30]. Endpoint reference is a dynamic alternative to the static service element defined in the WSDL since it allows redefining the service location dynamically.

WSDL describes services as collections of ports supporting the abstract definitions of service operations and the messages involved. This allows BPEL to discover dynamically some of the information contained in port definitions based on the abstract definitions in port type. That makes it possible to design the process based on an abstract definition of web services that are later determined over time, but it provides no prescription of what is to be done when the partner that is supposed to provide an endpoint reference fails. This sounds a perfect solution for the cases where the abstract level of information about operations and message involved are clear at design time. However, in most e-Procurement scenarios, trading parties are specified later over time.

One solution to utilize the advantage of endpoint references in e-Procurement applications is to design a private registry for services where all the interested suppliers can register their services. This private service registry is managed and verified by the organization. This service registry can later provide the required abstract information.

Karastoyanov et al. [31] provides a similar solution which focuses on a three level service discovery by (1) connecting to a standard UDDI registry followed by (2) selecting the suitable web service based on the selection policy and finally (3) binding the selected port. However, their approach suffers from a lack of proper selection policy. Moreover, using UDDI is not a suitable solution for e-Procurement since the service registry is typically a private registry of verified providers.

Chen et al [9] proposes an architectural design for web service enabled procurement through a private supplier web services registry database. Although it supports web service discovery, it has not addressed the mapping issues in binding discovered web services to e-Procurement processes. Hernandez et al. [32] discusses this issue at a higher level. They provide a web service-based brokering service for e-Procurement in supply chains. Although a dynamic binding service is a component in their proposed architecture that binds compatible business processes described as web services, no architecture has been provided for how to bind supplier services. Karastoyanov et al. [31] believes that future trends in this respect are towards the creation of a sophisticated infrastructure to support advanced dynamic features of service-oriented e-Procurement. Most solutions have suffered from a lack of support for dynamic service binding in BPEL defined e-Procurement processes. In this paper, we address this by proposing a framework for dynamic acquisition of procurement services in a continuously changing environment.

3 Dynamic Service Binding in E-Procurement

The definition of e-Procurement from [33] is adopted in this paper. We define e-Procurement as an electronic acquisition of goods and services. The e-Procurement process from a consumer's point of view includes a number of pre-purchase activities and after purchase activities. Pre-purchase activities cover:

1. Search for vendors and products
2. Qualify vendors
3. Select a market mechanism
4. Compare and Negotiate
5. Make a purchase agreement

After-purchase activities include:

1. Initiate a purchase order
2. arrange a pickup or receive shipment

3.1 E-Procurement Scenarios

Procurement in the health care industry presents a good scenario representative of a dynamically changing environment with dynamic procurement needs. This includes the procurement of healthcare professionals, medications, and healthcare services. For instance, procuring a room in the hospital for a surgery or the cost associated with the team of doctors and other healthcare professionals that will be involved in the procedure present other potential procurement scenarios. Important characteristics of these scenarios include:

- 1) they are event-driven,
- 2) service providers and consumers are constantly changing,
- 3) the rules of association is quite dynamic, and
- 4) the roles of each party is constantly changing.

Automating procurement processes in this industry present unique challenges since current standards do not support such dynamic service relationships. In this paper, we propose a framework for standardizing dynamic e-Procurement processes.

We use the health care e-Procurement automation as a good case study to illustrate our framework. In this scenario, we focus primarily on the procurement of medications. Pharmacies provide patient prescriptions from approved doctors. In doing this, the pharmacies interact with insurance companies on behalf of their patients to ascertain the level of coverage for each medication that is part of a patient's prescription. It is worth noting that different pharmacies may offer different prices for medications as well as different dispensing fees. These costs affect the patient choice of the pharmacy to deal with. In addition, the method of calculating the contributions and the procedure for obtaining a reimbursement vary from one insurance plan to another.

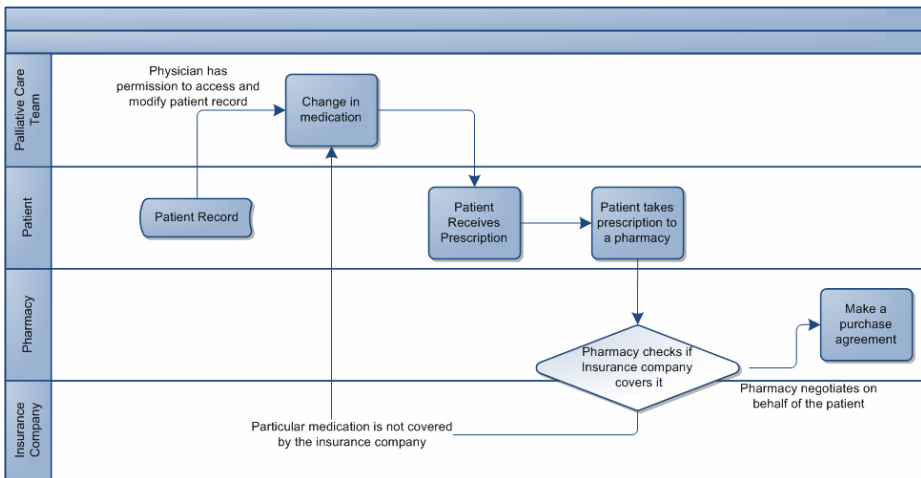


Fig. 1. Health care medicine procurement

Fig. 1 provides a structured process representing our scenario. It covers the process of medication prescription from the prescribing physician to the time the patient collects the medication from the pharmacy and the pharmacy is paid by the insurance company.

3.2 The Proposed Architecture

The e-Procurement processes identified in this scenario include:

1. Search for service providers
2. Qualifying providers

3. Determining a selection mechanism
4. Comparing and Negotiating services (if required)
5. Making an agreement

Since our interest is on procurement not transactions, we will focus more on the pre-purchase phase than the after-purchase and payment activities. To support the above definition, our proposed framework uses a message broker as a middleware for integrating potential suppliers of healthcare services. The message broker provides a registry service for all collaborating services while subscription policies define rules of collaboration in the B2B network [34].

The policy-based message broker is responsible for service discovery and acquisition of services. Once potential providers are specified, one might need to make procurement decisions through a decision making engine defined as BPEL processes. For example, a prescription may be covered partially by patient’s insurance plan with one insurance provider but fully covered by patient’s husband insurance plan with another insurance provider. Besides, there might be even some scenarios that required a few levels of negotiations between consumer and service provider to choose the final service provider.

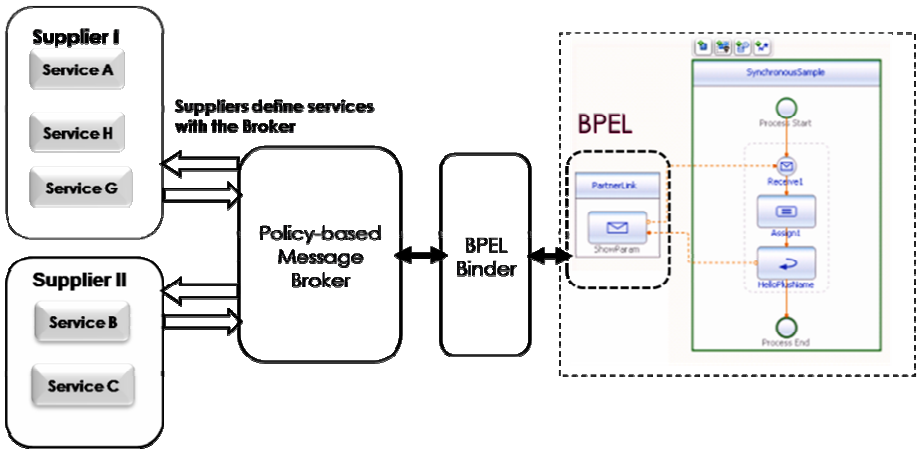


Fig. 2. Representing a collaboration between a BPEL and event-driven services

BPEL scripts provide a very flexible mechanism for describing and implementing these decisions. Negotiation and decision making could be considered as a separate component of architecture [35] or could be defined by BPEL as one of the internal steps of the procurement process [36]. Since participants may change by time, we prefer to consider decision making as one of the internal activities of the e-Procurement process defined by BPEL. Unfortunately, it is very difficult to support multiple service providers as required in e-Procurement negotiation processes. Our framework works around this by implementing special Partner Service Components[34] that interact with a BPEL process as *PartnerLink* but uses policy-based subscriptions to interface to many services providing similar services using an

event-driven publish/subscribe interaction pattern. This component is represented by the BPEL binder in our framework. Partners can join and leave the e-Procurement system without affecting the BPEL processes representing procurement negotiations, decisions and actions.

As shown in Fig.3, the policy-based message broker connects 3 types of service providers and consumers in an event-driven adhoc B2B framework. Patients publish *prescription*. Based on the subscription of patient pharmacy, these data structures are then received by the pharmacy for the patient. The pharmacy then analyses the patient request and subsequently publishes *InsuranceClaim* on behalf of the patient. This claim is picked up by the insurance company for the patient. If the medications in the prescription match those that are covered for the patient, a *coverage* for the claim is published back to the pharmacy. Medication is then issued and the insurance is then billed.

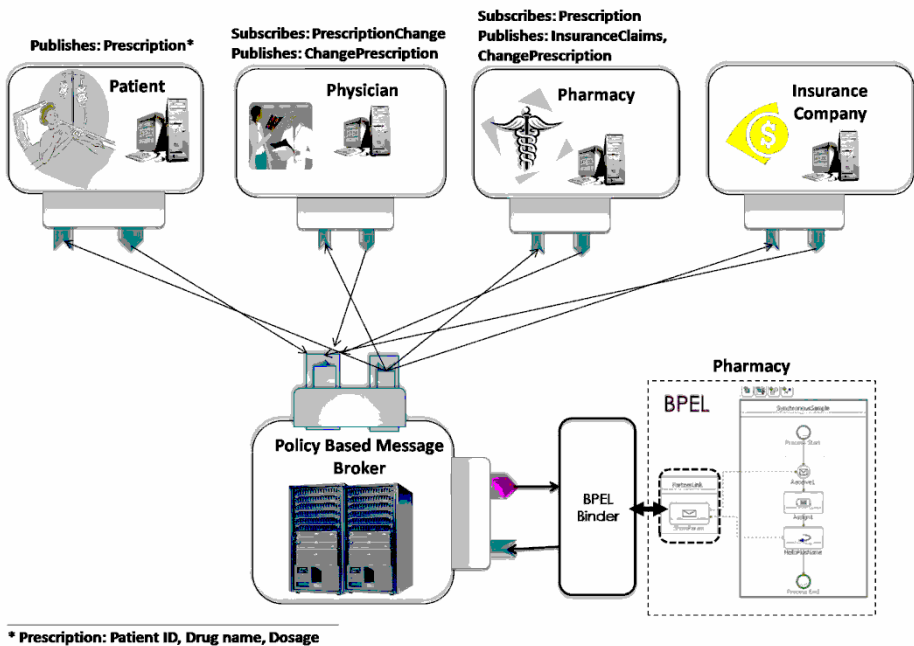


Fig. 3. Applying the Policy-based Message broker to e-Procurement

In cases where the *InsuranceClaim* is either not covered or partially covered by the insurance company, the insurance company may publish a *ChangePrescription* message that is then sent back to patient insurance and subsequently the patient physician for modifications and the process is repeated for the new prescription.

3.3 The Policy-Based Message Broker

The policy-based message Broker supports generic SOAP based interfaces, defined using WSDL, that a service can invoke in order to Register, Publish, and Subscribe as well as different options for receiving messages. Applications use a Partner Service

Component to communicate with the message broker. The component supports a SOAP interface, Notify, which the Message Broker can use to “push” messages to subscribers or, the Partner Service Component can define a “pullPoint” where the Message Broker will store messages until the Partner Service Component collects the messages using the `getMessages` interface.

The behavior of the Message Broker is controlled declaratively by policies stored in the policy database and a policy execution engine which executes them in response to message events. The Message Broker follows a totally policy-driven model to message handling and service interaction. Policies apply context to messages to determine what actions to perform on the messages (including formatting, filtering, storage) instead of hard-coding static point to point forwarding of messages. This allows incremental deployment of publishing and subscription functionalities as well as evolutionary refinement of message scheduling and formatting.

Messages are treated simply as structured XML documents that are not coupled to any specific operation. The subscriber and the broker identify the operation to be performed on a request or an acknowledgment by defining policies and using a policy engine to match the appropriate policies and actions to the context associated with the message.

4 Discussions of Results

WS-BPEL is oriented towards design-time binding of services. That is, at design-time, we need to know who the patient, the pharmacy, the insurance companies, and physicians are and the types of services they can provide. In an online environment, however, an e-Procurement process should be able to react and adapt in near real-time to changes in suppliers, requirements, and regulations. Our framework provides the architecture for achieving this. WS-BPEL is extended using a BPEL Binder that integrates with a Policy-based message broker to achieve the e-Procurement collaboration in an event-driven pattern[34]. The message broker infrastructure includes a policy engine administered by a policy administrator for organizing and maintain these rules for data sharing [34].

For evaluating our framework, we refer to a set of dimensions identified by Medjahed et al [14] to study interaction issues in B2B E-commerce. They consider coupling among partners, heterogeneity, autonomy, external manageability, adaptability, security, and scalability.

- **Coupling among partners:** it refers to the degree of tightness and duration of coupling among organizations. In our framework, loosely coupled partners exchange business information on demand where organizations need to dynamically discover partners to team up with to deliver the required service.
- **Heterogeneity:** heterogeneity refers to the degree of dissimilarity among organizations. Our framework supports applications using different internal data structures and standards as collaboration interfaces are web services.
- **Autonomy:** autonomy refers to the degree of compliance of a partner to global control rules. Our framework allows organizations to have more local control over implementation and operation of services, and flexibility to change their processes without affecting each other.

- **External manageability:** it refers to the degree of external visibility and manageability of partners' applications. With our publish/subscribe policy, organizations facilitate the prediction of their status and availability.
- **Adaptability:** this dimension refers to the degree to which an application is able to quickly adapt to changes. In an online environment, an e-Procurement process should be able to react and adapt in near real-time to changes in suppliers, requirements, and regulations. Our dynamic service binding solution effectively addresses this dimension.
- **Security:** Security must be enforced to give organizations the confidence that their transactions are safely handled. In this framework the media of interaction is over the web and each organization itself is responsible to provide the proper level of security to secure their web services. We have not focused on security on this paper.
- **Scalability:** scalability refers to the ability of a system to grow in one or more dimensions. Without a doubt, a low cost establishment of new relationships is desirable. Our framework with supporting dynamic binding effectively reduces this cost.

Although the current technologies provide the foundation for building dynamic integration frameworks, several research issues like security or privacy still need to be addressed to make these dynamic frameworks tangible.

References

1. Hawking, P., et al.: E-Procurement: Is the Ugly Duckling Actually a Swan Down Under? *Asia Pacific Journal of Marketing and Logistics* 16(1) (2004)
2. Yen, B.: *Migrating Procurement onto the Internet*, vol. 2, pp. 113–134. Kluwer Academic Publishers, Dordrecht (2002)
3. Abramson, M.A., Harris III, R.S.: *The Procurement Revolution*. Rowman-Littlefield (2003)
4. Thomas, E.: *Service-oriented Architecture: Concepts, Technology, and Design*. The Prentice Hall PTR, Englewood Cliffs (2005)
5. Ranjan, M., Dash, R.K.: *Suitability of Service Oriented Architecture for E-procurement. Technology in Government*. GIFT Publishing, New Delhi (2006)
6. SAP Discovery System - Service-Enabled Procurement Scenario, <https://www.sdn.sap.com/> (cited, February 2009)
7. Endrei, M., et al.: *Patterns: Service-Oriented Architecture and Web Services*, 1st edn. IBM Red Books, International Business Machines Corporation (2004)
8. Mahmoud, Q.H.: *Service-Oriented Architecture (SOA) and Web Services: The Road to Enterprise Application Integration (EAI)* (2005), <http://java.sun.com/developer/technicalArticles/WebServices/soa/> (cited, February 2009)
9. Chen, M., Meixell, M.: *Web Services Enabled Procurement in the Extended Enterprise: An Architectural design and implementation*. *Journal of Electronic Commerce Research* 4(10) (2003)
10. *Web Services Business Process Execution Language (WSBPEL)*, <http://www.oasis-open.org> (cited, February 2009)

11. Kimberley, P.: *Electronic data interchange*. McGraw-Hill, New York (1991)
12. Cobb, E., Batini, C.: The evolution of distributed component architectures. In: Batini, C., Giunchiglia, F., Giorgini, P., Mecella, M. (eds.) *CoopIS 2001*. LNCS, vol. 2172, pp. 7–21. Springer, Heidelberg (2001)
13. Lewandowski, S.M.: Frameworks for component-based client/server computing. *ACM Comput. Surv.* 30, 3–27 (1998)
14. Medjahed, B., et al.: Business-to-business interactions: issues and enabling technologies. *The VLDB Journal* 12, 59–85 (2003)
15. Glushko, R.J., Tenenbaum, J.M., Meltzer, B.: An XML framework for agent-based E-commerce. *Communication ACM* 42(3) (1999)
16. Microsoft Inc., BizTalk Framework Specification, <http://www.biztalk.org> (cited, February 2009)
17. cXML Version 1.2 Specification, <http://www.cxml.org> (cited, February 2009)
18. RosettaNet Specifications, Dictionaries, Implementation Framework, Partner Interface Processes, Maintenance Request Form, <http://www.rosettanet.org> (cited, February 2009)
19. ebXML Technical Architecture Specification v1.04
20. Levy, M., Homann, U.: Agreement and Organization: Protocol Architecture for B2B. Microsoft on-line articles (2004)
21. Chung, N.C.-N., Huang, W.-S., Tsai, T.-M.: eXFlow: A Web Services-Compliant System to Support B2B Process Integration. In: *Proceedings of the 37th Hawaii International Conference on System Sciences* (January 2004)
22. OASIS UDDI v. 3 Specification
23. Leymann, F.: *Web Services Flow Language (WSFL 1.0)* (May 2001)
24. Klein, M., Konig-Ries, B.: Combining query and preference - an approach to fully automatize dynamic service binding. In: *Proceedings of IEEE International Conference on Web Services* (2004)
25. Papazoglou, M., et al.: XSRL: An XML web-service request language. Technical Report DIT-02-0079, University of Trento (2002)
26. Fikes, R., Hayes, P., Horrocks, I.: DAML query language (DQL) - abstract specification, <http://www.daml.org/2003/04/dql/dql> (cited, February 2009)
27. Carey, S., et al.: SOA Best Practices: The BPEL Cookbook, Making BPEL Processes Dynamic. SOA Best Practices. The BPEL Cookbook
28. The ActiveBPEL Community Edition Engine, <http://www.activebpel.org/> (cited, February 2009)
29. Maynard, N., Akermann, H.: *Dynamic Service Binding with BPEL* (April 2006), <http://www.ibm.com/developerworks/webservices/library/ws-bpelwsad/> (cited, February 2009)
30. W3C Web Services Addressing (WS-Addressing), <http://www.w3.org/Submission/ws-addressing/> (cited, February 2009)
31. Karastoyanova, D., Houspanossian, A., Cilia, M.: Extending BPEL for Run Time Adaptability. In: *Proceedings of the Ninth IEEE International EDOC Enterprise Computing Conference (EDOC 2005)*. IEEE Computer Society, Enschede (2005)
32. Alor-Hernandez, G., et al.: A Web Service-Based Brokering Service for e-Procurement in Supply Chains. In: Cérin, C., Li, K.-C. (eds.) *GPC 2007*. LNCS, vol. 4459, pp. 686–693. Springer, Heidelberg (2007)
33. Turban, E., et al.: *Electronic Commerce 2008*. Prentice Hall, Englewood Cliffs (2008)

34. Eze, B.: A Policy-based Message Broker for Event-driven services in B2B networks, M.Sc. Thesis, University of Ottawa, Ottawa (2009)
35. Chandrashekar, T.S., Narahari, Y., Rosa, C.H., Kulkarni, D.M., Tew, J.D., Dayama, P.: Auction-Based Mechanisms for Electronic Procurement. *IEEE Transactions on Automation Science and Engineering* 4(3) (July 2007)
36. Benyoucef, M., Rinderle, S.: Modeling e-Negotiation Process for a Service Oriented Architecture. *Group Decision and Negotiation Journal* 15, 449–457 (2006)

Integrating Identity Management With Federated Healthcare Data Models

Jun Hu and Liam Peyton

School of Information Technology and Engineering,
University of Ottawa, Canada
{junhu, lpeyton}@site.uottawa.ca

Abstract. In order to manage performance and provide integrated services, health care data needs to be linked and aggregated across data sources from different organizations. The Internet and secure B2B networks offer the possibility of providing near real-time integration. However, there are three major stumbling blocks. One is to standardize and agree upon a common data model across organizations. The second is to match identities between different locations in order to link and aggregate records. The third is to protect identity and ensure compliance with privacy laws. In this paper, we analyze three main approaches to the problem and use a healthcare scenario to illustrate how each one addresses different aspects of the problem while failing to address others. We then present a systematic framework in which the different approaches can be flexibly combined for a more comprehensive approach to integrate identity management with federated healthcare data models.

Keywords: Federated data model, privacy, identity, record linking, data consolidation, and health care data.

1 Introduction

Performance management, knowledge discovery and integrated services are important for public health. They require linking and aggregating health care data across data sources from different organizations. Currently, in Ontario, this is done manually on a request basis (e.g. by phone to request a copy of paper-based records), or on a batch reporting basis (usually only for aggregated data). The Internet and secure B2B networks offer the possibility of providing near real-time integration. However, there are three major stumbling blocks. One is to standardize and agree upon a common data model across organizations. The second is to match identities between different locations in order to link and aggregate records. Often, there is no common identifier and the identifying information is inconsistent and often ambiguous. The third is to protect identity and ensure compliance with privacy laws [5, 4, 11]. Consent must be obtained, identity protected, and data sets must be de-identified. However, when linking data from different data sources, the combination of data may be potentially re-identified [1].

Our research is particularly interested in identity management for enabling federated data models, which can be used for data mining, performance management and

electronic health records. This involves matching identities from different data models to support linking and aggregating while at the same time protecting or hiding identity to ensure compliance with privacy laws. A number of different technologies and standards have been proposed to address this issue. In this paper, we survey three main approaches to the problem and use a healthcare scenario to illustrate how each one addresses different aspects of the problems while failing to address others:

- Three-Phase Consolidation Process Model
- Liberty Alliance Circle of Trust Model
- DB2 Anonymous Resolution Model

We then present a systematic framework in which the three approaches can be flexibly combined for a more comprehensive approach to integrate identity management into federated data models.

2 Healthcare Scenario

A federated data model is one in which separate, distributed, heterogeneous data is logically integrated to act as a single image. The component systems in a federated environment have significant autonomy in their execution, but they are willing to cooperate with others in executing user requests that access multiple data sources [10]. It is needed when organizations want to share data, but do not want a physical copy of their database to be created in any system that is out of their full control.

Fig.1 presents a health care scenario that concerns data in three different databases managed by different organizations. The clinic database is maintained by eClinic and records clinical data of patients such as symptom and diagnosis description. The

Simplified eClinic database and data.

Clinic table			Patient table			
Patient_id	Symptom	Date	Patient_id	Name	Birth Date	Health Card #
c1	High blood pressure	XXXXX	c1	John Wilson	1966-3-4	4444-222-333-AB
c2	Nausea	XXXXX	c2	Mark Weiss	1978-2-9	2354-256-167-DC
c3	Headache	XXXXX	c3	Joe Stern	1988-3-11	1234-789-732-AB

Simplified eHospital database and data.

Emergency Room table			Patient table			
Patient_id	Event	Date	Patient_id	Name	Birth Date	Health Card #
er7	Cardiac Arrest	XXXXX	er7	J Wilson	1.1ar 4. 1966	
er8	Stroke	XXXXX	er8	M Weiss	Feb 9. 1978	2354-256-167-DC
er9	Stroke	XXXXX	er9	Joe Stern	1.1ay 111988-	1234-789-732-AB

Simplified ePharmacy database and data.

Pharmacy Table			Patient table			
Patient_id	Prescription drug	Amount	Patient_id	Name	Birth Date	Health Card #
ph3	Drug1	100	ph3	Wilson, Jone	1966-03-04	4444-222-333-AB
ph4	Drug2	200	ph4	Mark Weiss	1978-02-09	2354-256-167-DC
ph5	Drug2	300	ph5	Joe Stern	1988-03-11	1234-789-732-AB

Fig. 1. Simplified healthcare databases in eClinic, eHospital & ePharmacy

database also contains patient’s profile information such as name and Health Card Number. The second database is the Emergency Room database maintained by eHospital and it stores the emergency encounter records, and the patient’s profile. The third database, a pharmacy database, is maintained by ePharmacy and it delivers a patient’s drug usage history and patient’s profile. We notice that there are inconsistencies in how the patients are identified in the three databases. For example, “John Wilson”, “J Wilson” and “Wilson, Jone” are the same person, in different database tables (including a misspelling of the first name).

The three databases are built and used separately. But, it would be useful if medical analysts could collect data on all patients from eHospital, ePharmacy, and eClinic to create a consolidated and composite view of a single patient like Fig.2. A data mining algorithm could be applied to this dataset to see if there are any patterns that might correlate Emergency Room visits for cardiac arrest with specific set of symptoms or diagnoses or prescriptions that might be in a patient's history. To do so, we need to integrate the data from three databases and resolve the inconsistency of patient’s identities while protecting identities. Therefore, a well defined identity management mechanism needs to be integrated into the federated data model so that healthcare data can be shared, linked and managed.

Consolidated Dataset

User	Event	Prescription	Amount	Symptom
1	Cardiac Arrest	Drug1	100	High blood pressure
2	Stroke	Drug2	200	Nausea
3	None	Drug2	300	Headache

Fig. 2. Consolidated dataset used for data mining

3 Existing Approaches for Integrating Identity Management Into Federated Data Models

In this section, we identify the technical issues that need to be addressed in our scenario, and examine existing privacy technologies and apply them to our federated data model. As part of this examination, we present an overview of the principles and merits of each approach as well as identify potential shortcomings.

3.1 Technical Issues for Identity Management in a Federated Data Model

Protection of personal information in multiple organizations entails addressing several requirements. Normally, in a single organization, data can be integrated from multiple data sources into a single data view using a common key. In a federated environment, though, the participant organization may restrict access to some data and demand to maintain full control of any access to the data. Moreover, a common unique identifier is often not available. In addition, the requested sensitive data (such as name, birth day, social insurance number, or health card number) is often prohibited from release. Even if it can be released, inconsistencies, duplicates and ambiguities are frequent.

Below are the main issues we have to address in identity management for sharing and linking healthcare data while protecting patient identities.

Depersonalization. The goal of depersonalization is to assure re-identification of a person is not possible. It can be achieved by removing, hiding and grouping sensitive data [12], or other statistical disclosure control (SDC) methods [15].

Identity linkage. In a federation environment, identities may be communicated and transformed in a variety of ways in each participant data warehouse. A mechanism is needed to allow these identities to be linked under appropriate circumstances. This is necessary when data mining needs to collect and integrate data from multiple sources. The identities can be linked directly, or anonymously, or pseudonymously, or deterministically, or probabilistically.

Anonymity and pseudonym. Anonymous and pseudonymous identities are two ways of protecting identity [8]. With anonymous identity, the identity is unknown and one cannot refer to an identity beyond the single session in which it is used. With pseudonymous identity, one can link events across sessions to an identity created for a specific organization, without knowing the actual identity. An important application of pseudonymity is that a person sustains separate relationships with multiple organizations, using separate identifiers, and generating separate data trails. These are designed to be very difficult to link, but, subject to appropriate legal authority, a mechanism exists whereby they can be linked.

Deterministic and probabilistic matching. Deterministic matching achieves a match only if the fields being compared are identical. This approach is suitable in a standardized and ideal federation environment. However, in a real world, there is no common identifier, and the identifying information is not standardized or consistent and often ambiguous. Probabilistic matching of identities is necessary in these situations. Probabilistic matching uses a greater number of matching variables to provide a maximum likelihood estimate among potential matches.

ID Consistency. ID data consistency is an important property when linking data from multiple sources. However, in healthcare, each system usually has its own user profile management system where identifiers are not necessarily consistent. It is better to have an identity management solution capable of handling id inconsistency.

Trusted third-party. Most secure systems need a trusted third-party organization to perform pseudonymization, record linkage, or encryption of the identities or other sensitive data.

3.2 Three-Phase Consolidation Process Model

The first approach to integrate identity management into federated healthcare data models we consider is the three-phase consolidation process model. This is typically used to provide mature decision support services for sophisticated research among healthcare organizations in a federated data warehouse. The conceptual model of this federated system is based on a widely accepted international standard to handle the heterogeneity of the components [12]. As shown in Fig.3, the three phases are

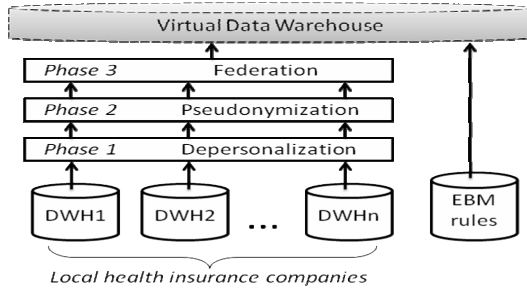


Fig. 2. Three Phases of Data Warehouse Consolidation ([12])

depersonalization, pseudonymization and federation. Depersonalization and pseudonymization are used to ensure that personal identities are made secret before sending data to be federated.

In the depersonalization phase, sensitive data and their sensitivity levels need to be specified while creating the conceptual federated data model. Depersonalization can be achieved by removing, hiding and grouping sensitive data. After completing the depersonalization process, the pseudonymization process can be performed by a trusted third-party organization. Pseudonymization transforms and masks identity. To enable linking partial result sets for a single patient, the encrypted unique patient identifiers must be passed to the third-party for pseudonymization. When a query is submitted to the federated data warehouse, it is reassembled into sub-queries for each of the underlying data sources. After depersonalization, encryption and pseudonymization, all partial query results are decrypted and consolidated into a single result query for the user.

However, this approach does not work well with the scenario introduced in section 2. In the depersonalization phase, Name is removed and Birth date is grouped. Patient_id cannot be an identifier since it is defined differently in different organizations. Health card number is a good candidate, but it is missing for a patient in eHospital. If each organization passes different identifiers to the third-party, the third-party in this approach cannot resolve this kind of id inconsistency. Therefore, the data from eClinic, eHospital and ePharmacy for a single patient will not be linked.

In summary, three-phase consolidation process has the following characteristics:

- 1) Depersonalization protects the personal info to some extent
- 2) Pseudonymization transforms and masks identity while linking partial result set.
- 3) Third-party transforms identity and sees all encrypted data for the user.
- 4) The identify info, personal data and health data must be encrypted for trusted third parties before performing pseudonymization process.
- 5) All component data sources must use the same consistent identity info, such as social security number, so that the partial results for a single individual can be consolidated.

Obviously it works well in which identifiers are available, consistent, and valid in all datasets to be combined, and integration simply joins on the basis of these

identifiers. In general, though, even if datasets contain identifiers, their equivalence across datasets of different data sources is not necessarily guaranteed. In such situations, other identifying characteristics (such as names, birth day or addresses of persons) have to be taken into account to resolve ambiguities. In such situations, the approach does not work.

3.3 Circle of Trust Model

Circle of Trust (CoT) is a key concept related to B2B networks that has been developed by the Liberty Alliance project [14]. Liberty Alliance is a consortium of technology enterprises who have created an open standard and set of specifications for federated identity management. A CoT is a B2B network in which an individual's identity and personal information is protected by a designated Identity Provider, while still allowing cooperating organizations within the CoT to access and share information about the individual in a systematic manner that ensures the individual's permission is obtained and their identity protected [8]. These cooperating organizations have trust relationships and operational agreements established amongst them. Pseudonyms are central to how identity is protected in a CoT.

A trusted third party Identity Provider in a CoT maintains a Client Master Index, which is a directory of users' pseudonyms and associated 'pointers' to the different service providers where their data records reside. Fig.4 shows a Client Master Index for the scenario in section 2. Pseudonyms in eClinic, eHospital and ePharmacy are assigned by the Identity Provider and maintained in the Client Master Index. Each of the three organizations recognizes the patient by a different pseudonym known only to them.

In addition to the Client Master Index, the Identity Provider manages identity information, also shown in Figure 4. Rather than each of eClinic, eHospital and ePharmacy maintaining duplicate and conflicting information in their own Patient Table, the Identity Provider is the only source of identity. For most processing, there is no need for identity information. When it is needed, it is sourced from the Identity Provider who ensures that proper permissions are in place. Using special encryption algorithms, the Identity Provider supports linking of data records from one service provider to another, when proper permissions are in place, through a discovery service. The architecture of a data collection service that could build a consolidated data set, like the one in our scenario, is described in [6]

In summary, when a CoT is leveraged to support a federated data model, it has the following characteristics:

- 1) Identity linkage uses a central pseudonym map (Client Master Index).
- 2) Identity Provider provides pseudonyms to link partial result sets.
- 3) Identity Provider maintains and protects a single, consistent set of identity information.

The crux of this approach is the effort and coordination required to set up a CoT among all participant organization, create an Identity Provider organization and have each individual in the network register with the Identity Provider, and then set a master index to link each individual accurately.

Client Master Index

User_id	Service	pseudonym
U1	eClinic	c1
U1	eER	er7
U1	ePharmacy	ph3
U2	eClinic	c2
U2	eER	er8
U2	ePharmacy	ph4
U3	eClinic	c3
U3	eER	er9
U3	ePharmacy	ph5

Identity Information

User_id	Name	Birthdate	HealthCard #
U1	John Wilson	1966-03-04	4444-222-333-AB
U2	Mark Weiss	1978-02-09	2354-256-167-DC
U3	Joe Stern	1988-03-11	1234-789-732-AB

Fig. 4. Identity Provider: Client Master Index and Identity Information

3.4 DB2 Anonymous Resolution Model

DB2 Anonymous Resolution (AR) [3, 13] is an IBM privacy technology solution to allow organizations to anonymously share and compare information. There are four components in the AR solution:

- 1) Pre-Processor performs standardization, correction, and normalization of the data elements;
- 2) Anonymizer applies a one-way hash function [9] to transform information into cryptographic values that are impossible to mathematically convert back to their original form;
- 3) Resolver processes anonymized data to identify entity matches, detect relationships and generate alerts based on identified matches or relationship;
- 4) Console manages AR configuration and alerts.

The anonymization process removes all personal identity information, leaving only a pointer to the original information. It leverages pre-processing techniques before the one-way hash is applied. As a result, AR achieves a sort of fuzzy matching to recognize ambiguities, misspellings, or partial records within a data set [7]. For example, in the previous scenario, AR can recognize that “John Wilson”, “J Wilson” and “Wilson Jone” are the same patient based on their name, birth date and health card number. In summary, when AR is applied to federate data model, it has the following characteristics:

- 1) Third-party helps deal with data quality issues and maintains encrypted identity information.
- 2) Third-party does not see other data except encrypted identity info.
- 3) Identity linkage uses a central encrypted identity information map.
- 4) The identity data may vary at each component data source.

AR adapts probabilistic matching techniques and depends largely on the completeness and accuracy of the information to be linked and an appropriate combination of matching variables. Therefore data quality and process order is extremely important for this solution. It accepts an inconsistent, ambiguous state and does the best to

match, but it cannot guarantee 100% accuracy. This can result in an unreliable anonymous federated data model.

3.5 Summary of Approaches

In Table 1, we compare the three approaches with respect to the technical issues identified in section 3.1.

Each of the three approaches focuses on different aspects of identity management. The three-Phase consolidation model focuses on de-identification. It fails to satisfy a more generic need when there is no common identifier across organizations. The Liberty Alliance Circle of Trust uses a system of pseudonyms, managing identity information and allowing the sharing of data between organizations. DB2 anonymous resolution resolves ambiguous, anonymized data to identify potential identity matches. In short, the existing systems may provide powerful solutions for some aspects of the problems describe in section 1 while failing to address others.

Table 1. Comparison of three approaches

Issue	Three-phase Consolidation	Circle of Trust	Anonymous Resolution
Depersonalization	Remove, hide and group sensitive data	Separate identity data from other data	One-way hash to hide identity
Pseudonym	Yes	Yes	No
Anonymity	No	No	Yes
Identity linkage	Pseudonymously	Pseudonymously, Client Master Index	Anonymously
Matching technique	Deterministic matching	Deterministic matching	Probabilistic matching
ID Consistency	All data sources must use the identical IDs and ID info	Identity Provider for ID and ID info, data sources linked via pseudonym map.	ID and ID info varies at each Data source, use a ID info hash
Trusted third-party	Sees all pseudonyms, encrypted ID info, and encrypted data	Sees pseudonyms map. Sees ID info. Sees no other data.	See encrypted ID map and ID info, but no other data.
Shortcoming	Fails when no common identifier across organizations.	Complex to set up CoT.	Unreliable to map identities.

4 Systematic Framework for Integrating Identity Management into Federated Healthcare Data Models

In this section, we present a systematic framework in which the different privacy technologies can be flexibly combined for a more comprehensive approach to integrate identity management into federated healthcare data models that can be used for knowledge discovery, performance management and electronic health records.

4.1 Framework Overview

Our goal is to share, link and protect identity in federated healthcare environment to enable integrating healthcare data from different organizations. The identity management in our proposal is based on the concept of Circle of Trust, where identification of patients through a Master Patient Index (MPI) is maintained by a trusted third party Identity Provider. A MPI creates patients' pseudonyms and associated pointers to the different organizations where their healthcare data records reside. It enables a consolidated, composite view of a single patient in federated healthcare data model.

Generally the main steps of data integration in federated healthcare data models include preparing data for sharing, consolidating data from multiple data sources, and evaluating consolidated dataset. Based on it, the proposed framework for integrating identity management also includes three main components as shown in Fig.5.

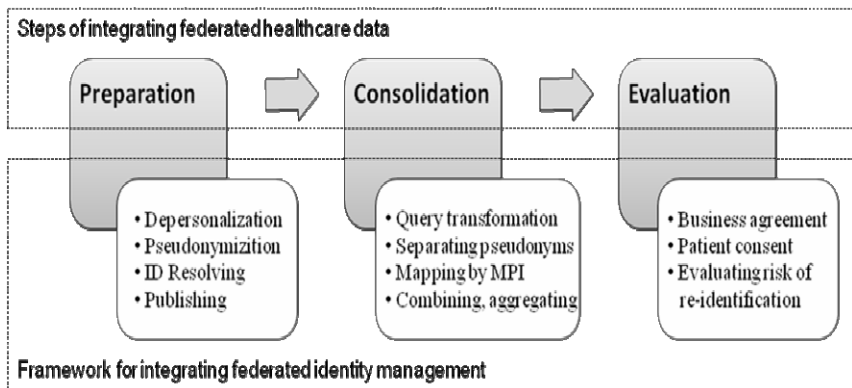


Fig. 5. Main components of the framework for integrating identity management

4.2 Preparation of Data for Sharing

In order to be shared in the Circle of Trust, a data source has to have its own pseudonym linked to the Master Patient Index and permissions for sharing. Three phases in the data preparation process are used to manage patient's identity and personal data:

- 1) Depersonalization phase removes, hides, or groups sensitive data.
- 2) Pseudonymization phase creates or updates pseudonyms linked to the MPI.
- 3) Publishing phase registers the data source for sharing.

In the depersonalization phase, we need to recognize the sensitive data, identify its sensitivity level, and specify the depersonalization measure. For example, patient name is very sensitive and it should be removed. Date of birth is in the middle sensitivity level, and we can create a new attribute "age range" to record this info.

In the pseudonymization phase, Anonymous Resolution (AR) can be used for potential matches of existing identities when adding a new data source or first preparing an old data source to the Circle of Trust. AR technologies can resolve duplicates, ambiguities and inconsistencies such as in Fig.1 where patient name and address are not consistent in the three different databases.

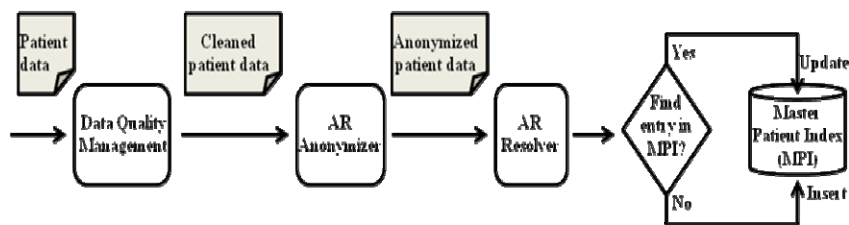


Fig. 6. Mapping identities and updating a MPI using Anonymous Resolution

Fig. 6 is a process of mapping identities using AR. If AR finds that there is not an existing identity associated to the data source, then the Identity Provider inserts an entry to the MPI, and assigns a pseudonym to this data source. Otherwise, the Identity Provider will update the entry associated to the existing identity in the MPI.

4.3 Consolidation in Federated Data Model

After the data source is published in the Circle of Trust, identity management reconciles the pseudonyms and aggregates data to develop a consolidated view of the patient that can be used for performance management, knowledge discovery and etc. Fig. 7 shows a privacy-preserving query process in a federated data model. The steps of the query are as follows:

- 1) A user query is submitted to the federation site.
- 2) The query is reassembled into sub-queries by checking the Dataset Registry ensuring that all approvals have been obtained
- 3) Each underlying organization processes its corresponding query. The partial query result includes two parts. One is pseudonym and the other is the required healthcare data.
- 4) The pseudonym part is sent to the Identity Provider.
- 5) The other healthcare data is sent to the federation site.
- 6) The Identity Provider uses The MPI to convert pseudonyms for federation, and send them to the federation site.
- 7) The federation site consolidates all partial query results into a single result query that is delivered to the user.

4.4 Evaluation of Consolidated Dataset

Evaluation is the process to determine if the identity and personal data are protected in the consolidated process and consolidated dataset. In addition, evaluation needs to check if there still exist some important identity management and privacy issues that need to be further considered. The following is the issues that identity management must be addressed:

- 1) Appropriate consent is obtained from patients for collecting their data.
- 2) Business agreements are in place both for obtaining the data from participant organizations and sharing the results.

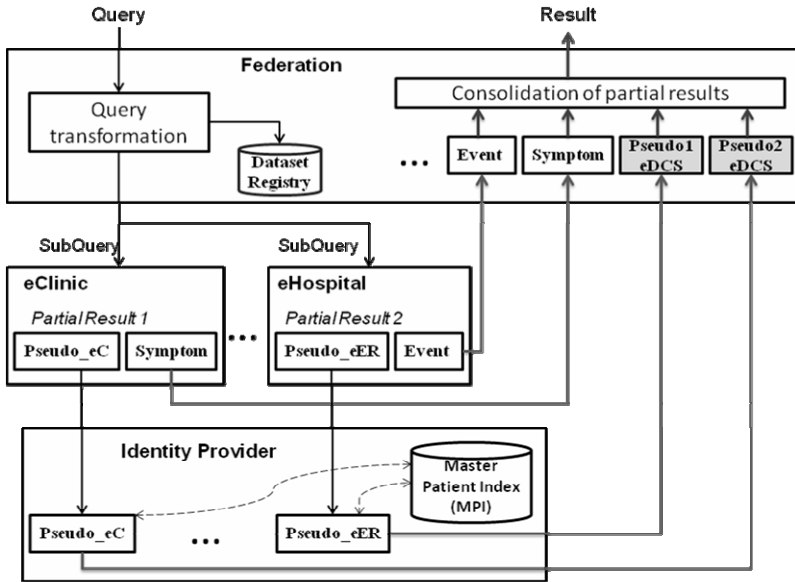


Fig. 7. Integrating identity management in a query process

- 3) The query users or services have the required access rights at each organization for the data required.
- 4) The consolidated dataset will not create identifiable data.

We use the techniques introduced in [1] to measure the risk of re-identification for the consolidated dataset. The dataset can be evaluated by using estimation methods such as data intrusion simulation [2] to predict the probability that a randomly selected patient can be matched successfully using a match process that an attacker would likely use in order to re-identify a de-identified dataset.

5 Conclusions

In this paper, an e-health scenario was first introduced to define the technical issues related to identity management that have to be addressed for enabling federated data models that can be used for performance management, knowledge discovery, electronic health records, etc. Then we analyzed existing privacy technologies and identified their characteristics and potential shortcomings when they are applied to a federated healthcare data model. Finally we proposed a systematic framework in which the different privacy technologies can be flexibly combined for a more comprehensive approach to integrate identity management into federated healthcare data models.

Our proposal is based on the concept of the Circle of Trust where each data source has its own pseudonym linked to a Client Master Index and permissions for sharing. Three phase approach is used to prepare data for sharing in a Circle of Trust.

Anonymous Resolution (AR) is used in a data preparation stage when first preparing an old data source and adding a new data sources into the Circle of Trust. AR can spot ambiguities, duplicates, and can look for potential matches of existing identities in the Circle of Trust. Finally techniques for measuring the risk of re-identification are used in evaluating of the consolidated dataset.

Acknowledgements

This work was partially supported by a Collaborative Health Research Project grant from CIHR and NSERC (Canada) on Performance Management at the Point of Care: Secure Data Delivery to Drive Clinical Decision Making Processes for Hospital Quality Control.

References

1. El Emam, K., Jabbouri, S., Sams, S., Drouet, Y., Power, M.: Evaluating common de-identification heuristics for personal health information. *Journal of Medical Internet Research* 8, e28 (2006)
2. Elliot, M.: A new approach to the measurement of statistical disclosure risk. *International Journal of Risk Management* 2(4), 39–48 (2000)
3. Friedrich, A.: IBM Entity Analytic Solutions, IBM DB2 Anonymous Resolution: Knowledge discovery without knowledge disclosure, IBM DB2 Anonymous Resolution Whitepaper (May 2005), <ftp://ftp.software.ibm.com/software/data/pubs/papers/db2anonymousres.pdf> (last accessed, January 2009)
4. Government of Ontario: Personal Health Information Protection Act (2004), http://www.elaws.gov.on.ca/html/statutes/english/elaws_statutes_04p03_e.htm (accessed, January 2009)
5. Health Insurance Portability and Accountability Act, United States Congress, United States, <http://aspe.hhs.gov/admsimp/pl104191.htm> (last accessed, January 2009)
6. Hu, J., Peyton, L., Turner, C., Bishay, H.: A model of trusted data collection for knowledge discovery in B2B networks. In: *The 2008 International MCETECH Conference on e-Technologies*, pp. 60–69. IEEE Press, Washington (2008)
7. Jonas, J.: Threat and Fraud Intelligence. Las Vegas Style, *Security & Privacy Magazine* 4, 28–34 (2006)
8. Koch, M., Möslein, K.M.: Identity Management for Ecommerce and Collaborative Applications. *International Journal of Electronic Commerce* 9(3), 11–29 (2005)
9. Naor, M., Yung, M.: Universal one-way hash functions and their cryptographic applications. In: *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, Seattle, Washington, United States, pp. 33–43 (1989), ISBN:0-89791-307-8
10. Ozsu, M.T., Valduriez, P.: *Principles of Distributed Database Systems*, 2nd edn. Prentice-Hall, Upper Saddle River (1999)
11. PIPEDA, Government of Canada, Health Information Custodians in the Province of Ontario Exemption Order, <http://canadagazette.gc.ca/partII/2005/20051214/html/sor399-e.html> (accessed, January 2009)

12. Stolba, N., Banek, M., Tjoa, A.M.: The Security Issue of Federated Data Warehouses in the Area of Evidence-Based Medicine. In: The First International Conference on Availability, Reliability and Security, pp. 11–22. IEEE Press, Washington (2006)
13. Swire, P.: Research Report: Application of IBM Anonymous Resolution to the Health Care Sector,
http://www.ehcca.com/presentations/cclf3/swire_s5_t4.pdf
(last accessed, January 2009)
14. Wason, T. (ed.): Liberty ID-FF Architecture Overview; version 1.2, Liberty Alliance Project, New Jersey (2003),
http://www.projectliberty.org/liberty/resource_center/papers
(last accessed, January 2009)
15. Willenborg, L., de Waal, T.: Elements of statistical disclosure control. Springer, Heidelberg (2001)

Wrestling With a Paradox: Complexity in Interoperability Standards Making for Healthcare Information Systems

Jeff Pittaway and Norm Archer

McMaster University, DeGroote School of Business,
1280 Main Street, Hamilton, Ontario, Canada L8S 4M4

Abstract. Medical interventions are often delayed or erroneous when information needed for diagnosing or prescribing is missing or unavailable. In support of increased information flows, the healthcare industry has invested substantially in standards intended to specify, routinize, and make uniform the type and format of medical information in clinical healthcare information systems such as Electronic Medical Record systems (EMRs). However, fewer than one in four Canadian physicians have adopted EMRs. Deeper analysis illustrates that physicians may perceive value in standardized EMRs when they need to exchange information in highly structured situations among like participants and like environments. However, standards present restrictive barriers to practitioners when they face equivocal situations, unforeseen contingencies, or exchange information across different environments. These barriers constitute a compelling explanation for at least part of the observed low EMR adoption rates. Our recommendations to improve the perceived value of standardized clinical information systems espouse re-conceptualizing the role of standards to embrace greater flexibility in some areas.

Keywords: Electronic medical record adoption, technical issues, social issues, interoperability, complexity, standards, standards making.

1 Introduction

Medical interventions are often delayed or erroneous because information needed for diagnosing or prescribing is missing or unavailable, and the consequence is reduced health outcomes and lost human lives [22, 23]. Therefore, when patients interact with different healthcare practitioners, such as physicians, specialists, hospitals, and medical labs [4], the information recorded by each healthcare practitioner should be available to other practitioners that treat the same patient over time [20]. To that end, many jurisdictions are implementing information technology (IT) networks over which healthcare systems can exchange information. For example, the federal and provincial governments of Canada have invested heavily in secure networks and an Electronic Health Record (EHR) architecture intended to integrate patient medical records from physicians, hospitals, public health agencies, medical laboratories, diagnostic images and pharmacies [2, 34]. By supplying information needed for diagnosing or prescribing, EHRs are expected to reduce delays and errors in medical care [22, 23]. The

achievement of these benefits depends on physicians accessing information from and supplying information to these systems. Physician access is primarily facilitated by linking the Electronic Medical Record (EMR) systems that many physicians use to manage clinical records. Therefore, the degree to which important medical information is exchanged electronically among healthcare practitioners depends in large part on physician adoption of EMR systems. However, at this point fewer than one in four Canadian physicians have adopted electronic medical record (EMR) systems. As a consequence of poor EMR adoption, patient records are often fragmented among doctors' offices, clinics, test centres, labs, and hospitals. When patients are transferred among healthcare facilities, fragmented electronic records can result in errors or mislaid information, unnecessary delays, duplication of effort, higher costs, and reduced quality of care [4, 15, 22, 23].

The causes of poor EMR adoption have been well studied from the perspective of technical integration issues [6, 15, 20, 21, 38]. In the technical perspective, information flow increases when more healthcare practitioners are integrated through information systems that can exchange information. Conversely, incompatibilities between systems cause the fragmentation of information. However, even when systems are technically compatible, information flows among healthcare systems fail if different healthcare practitioners use different definitions, terms, identifiers and rules to record data. To mitigate the problem of heterogeneous data, there is a widespread effort to standardize records and protocols. The objective of standards is to specify, routinize and make uniform the type and format of medical information to ensure that a medical observation, for example, can be recorded by one healthcare professional, forwarded to another healthcare system, and interpreted by another healthcare user in another facility at a different time with consistency [17]. The standards perspective, therefore, emphasizes that predefined standards should be embedded in healthcare information systems in order to support consistent information flows among healthcare practitioners. However, our contention in this paper is that the standards making process faces important barriers that constitute a compelling explanation for low adoption of EMR systems by physicians. Therefore, the purpose of this paper is to understand barriers to standards making in order to re-conceptualize the role of standards in improving information exchange among healthcare practitioners.

The remainder of this paper proceeds as follows. First, we review the state of EMR adoption in North America and Europe. Healthcare system interoperability is then discussed, emphasizing the importance of interoperability in unlocking the value of medical records distributed among healthcare practitioners. We then review the nature of the standards making process and the barriers to standards making constituted by technical complexity, social complexity and equivocality in standards making. Finally, we recommend two approaches to improve the value of standards to healthcare practitioners in the quest to increase practitioner adoption of systems for sharing essential medical records.

2 Electronic Medical Record (EMR) System Adoption by Physicians

Each year in Canada, more than 100 million physician examinations and 500 million laboratory and radiology tests are performed, and 382 million prescriptions are written

[31]. Yet, many healthcare practitioners still operate with paper-based information flows or heterogeneous and disconnected IT systems [9, 16, 23]. As discussed previously, poor adoption of EMR systems among healthcare practitioners leads to delayed or erroneous medical interventions because information needed for diagnosing or prescribing is missing or unavailable [22, 23]. However, adoption of EMRs by healthcare practitioners is alarmingly lower in North America than in Europe.

The European Union leads the world in electronic health adoption, and is moving ahead with the development of interoperability across national boundaries with the goal of achieving an effectively interoperable system by 2015 [13]. "Europe currently has a leading position in the world, with patient data being stored electronically by 80% of all EU-wide primary care physicians. About 70% of European doctors use the Internet and 66% use computers for consultations. Administrative patient data is electronically stored in 80% of general practices: 92% of these also electronically store medical data on diagnoses and medication" [12, p. 1].

While the United States exhibits significant initiatives and some progress in healthcare systems adoption [25], a U.S. study [3] reveals that just 20% to 25% of primary care physicians report using EMR systems and one third of physicians do not perceive any benefits of IT. In a recent national survey of Canadian hospitals [36], slightly more than half (54.2%) reported having some sort of EHR in place. However, 97.6% indicated that the EHR was not the sole method for recording patient information and thus records are fragmented. Furthermore, EMR use by primary care physicians varies substantially among Canadian provinces [7]. For example, the Province of Alberta's POSP (Physician Office Support Program) initiative that subsidizes and supports EMR adoption had enrolled just 61% of Alberta physicians as of May 2006 [32]. In contrast, OntarioMD, a subsidiary of the Ontario Medical Association has enrolled just 1,200 of the 17,500 physicians in that province [30]. Consequently, patient records remain mostly in paper form scattered among physicians' offices, clinics, test centres, labs, and hospitals.

A range of factors theorized to impact EMR adoption by primary care physicians are the topic of a recent literature survey [5]. One theme emerging from the literature is that physicians do not perceive the risk of transitioning to IT systems to be outweighed by the perceived benefits. Risk arises in adopting EMRs because existing paper records are expensive to convert to electronic form, office practices must be adapted to clinical system processes, staff must be retrained, systems must be acquired and implemented, and the changeover must be managed carefully in order to avoid undue disruption to the work being performed. Approximately 50% of such transitions have been unsuccessful [21]. Therefore, to increase EMR adoption requires that physicians perceive value from integrated EMR systems that outweighs the risks of integrated EMR adoption; they must perceive substantial value from EMRs over paper-based and antiquated IT systems. Much of the information EMRs can provide of value to physicians is only available from external sources (e.g., lab tests, diagnostic image scans, specialist reports, information on events and treatments at institutions such as critical care hospitals) [4]. Therefore, the ability to assemble all the information relevant to a patient's case in an electronic form so it can be manipulated, analyzed, and presented in support of decision-making is essential to the viability and value of EMRs for physicians. The healthcare industry has, therefore, come to recognize the critical need for interoperability among information systems that maintain health records.

3 Healthcare System Interoperability

Of major importance to interoperability in healthcare is the uniform movement of data from one system to another such that the clinical or operational purpose and meaning of the data is preserved and unaltered [19]. If this is to be accomplished, the users of these systems need to agree upon a standard format for recording and exchanging medical information. Information exchange among healthcare systems fails when different healthcare practitioners use different definitions, terms, identifiers, file formats, and data rules to record information [38]. Thus, the importance of standards in enhancing electronic information flows and value to physicians has been the topic of several studies, as follows.

Researchers have often found that the absence of standards constitutes a barrier to interoperability. Nøhr and Boy [29] present empirical evidence from a longitudinal survey in the Danish Ministry of Health wherein regional healthcare administrators rated the lack of standards as the second largest barrier to EMR adoption. Goodhue *et al.* [14] found that the lack of data standards makes it difficult or impossible to share or interpret data across systems. Bergeron and Raymond [6] found that the use of pre-defined transmission standards and protocol is a significant factor in the failure of enterprise information systems. In an empirical study of 111 enterprises, Wixom and Watson [37] found that practitioners significantly associated the existence of common definitions for key data items with implementation success of systems that aggregate large volumes of data. Lack of commonly defined data elements and codes across information systems also hinders users from enacting coordinated responses to problems [15]. Hanseth *et al.* [17] interpret failures to adopt EMR systems as an inherent outcome of problems with the standard making process. Thus, in order to analyze the problems with interoperability standards in healthcare, we must first understand the nature of the standards making process and what drives it.

3.1 Standards Making in Healthcare

Standards making is driven by (1) a need to manage interdependencies between healthcare practitioners, and (2) a need for better information to resolve uncertainty in the administration of care. Interdependencies exist between healthcare practitioners that must rely on other practitioners for information essential to care decisions. Therefore, a positive relationship has been suggested between interdependence and standards [27, 38]. This rationale asserts that when healthcare practitioners are highly interdependent and cannot satisfy their information dependencies because of semantic inconsistencies, they are driven to develop standards to overcome the problem. Uncertainty – referring to the absence of specific, needed information to support decision-making or administration – is also a positive driver of standards making [15, 38]. The rationale for this assertion is that when healthcare practitioners cannot support decisions or actions due to insufficient information to overcome uncertainty, they are driven to develop standards to overcome the problem. Standards making in healthcare is the mandate of standards development organizations (SDOs) that include, among others, practitioners, regulators and researchers.

In the process of standards making, SDOs usually adopt a reductionist approach [17]: a philosophical position that a complex system is nothing but the sum of its parts

and that an account of it can be reduced to accounts of individual constituents. However, Hanseth *et al.* [17] show that the problem is far more complex and that an overly reductionist approach to standards making can make the process self-destructive. These insights are presented in the following section where we analyze two sources of complexity and their implications on the standards making process in healthcare. One perspective analyzes the technical sources of complexity in standards making and illustrates that the panacea of universal standards may not be achievable through overly reductionist approaches. The second perspective analyzes social sources of complexity where the collaboration, cooperation and motivations of standard-makers take standards making on trajectories that may not achieve the original goal of the standard.

3.2 Technical Complexity as a Barrier to Standards Making

Schneberger and McLean [33] developed a definition of system complexity arising from the number of different types of components, the number of types of links, and the speed of change of the system. System complexity is also increased by the manner in which the system elements interact in a dynamic and nonlinear fashion, forming loops and recurrent patterns involving both positive and negative feedback [8]. Standards making in healthcare is thus considered highly complex because of the sheer volume of terms, data and information systems that must be ultimately managed. For example, the goal for universal healthcare records implies the daunting feat of tracking all health-related interventions and related data from cradle to grave for hundreds of millions of patients while addressing privacy and confidentiality standards and linking tens of thousands of information systems [20]. The accelerating pace of technical change affecting the supporting systems further amplifies complexity [17]. Even on a localized scale, the number of technical variables and interactions among these variables is high.

In practice, healthcare practitioners find that pre-defined standards and protocols do not work in all circumstances and in all localized conditions [35]. Given “non-standard” local circumstances, standards are too restrictive and interfere with the provision of care. Consequently, healthcare practitioners selectively appropriate standards they perceive to fit in certain circumstances, and adapt or work around protocols they perceive as too restrictive. With certain patients, for example, practitioners will deviate more from prescribed standards than others, or may combine the guidelines of several protocols (e.g., when patients suffer from multiple conditions simultaneously). Therefore, standards making in the technical perspective is a process of attempting to simultaneously balance localized information needs and work practices with a universal conceptualization of terminology, data and protocols, and embed both into explicit standards [17, 26].

Developers of healthcare information systems, in turn, attempt to embed standardized terminology, data items, rules and protocols in systems such as EMRs. However, developers find that healthcare “standards” such as HL7 have spawned many taxonomic branches in an attempt to accommodate localized information needs. Within each branch, the “standard” for recording a patient profile, for example, has been redefined, the multiple definitions are inconsistent and often incompatible, and developers are left arguing over how to implement standards [39, 40]. Furthermore, the

potential for overlap and inconsistency is built into the governance structures of some standards. For example, the International Health Terminology Standards Development Organization (IHTSDO) requires vendors that wish to use the SNOMED CT standard in their healthcare information systems to be licensed. However, the license grants licensees the right to “create extensions and derivatives from the International Release and use and modify those extensions and derivatives” [28, clause 2.1.2]. Developers using this clause can thus be directed by local healthcare customers to embed localized “versions” of a patient profile, for example, rather than implement a universal standard. Rather than leading to universality, these actions exacerbate the complex landscape of interoperability standards.

In the attempt to balance localized information needs and practices with the need for universality, unintended side effects of making one change to a standard reflect back onto the standards body, thus precipitating a modification to the standard, which creates side effects again, and so on. This cycle is known as reflexivity [17], and it amplifies the complexity of managing the many numerous and interdependent variables in healthcare. Hanseth *et al.* [17] relate an example of what at a cursory glance may seem a simple matter: developing standards based on a paper form currently utilized in practice. The researchers found that building standards from such artifacts proves, instead, to create a self-perpetuating cycle of failures, re-specifications, negotiations, changes, and so on. Ultimately, the record became so complex to accommodate all localized and situated needs that users found it difficult to find the specific information they needed in records. To cope with the cycle, practitioners began to make subjective choices to comply or not comply with the standard, and even to implement workarounds. In this case the reductionist approach to a simple form ultimately increased rather than reduced complexity.

Thus, the reductionist approach of describing standards in finer and finer details in the quest to achieve a universal standard tends, paradoxically, toward a self-destructive spiral rather than toward achieving universality. While the preceding focuses on complexity arising from the number of technical variables and interdependencies, social aspects of the process amplify the complexity of standards making further.

3.3 Social Complexity as a Barrier to Standards Making

Actors within SDOs, the practitioner actors they seek to standardize, and the information technologies employed by these practitioners constitute an actor network [17]. The term ‘actors’ refers to participants that process information in the administration of healthcare. Networks of actors develop formally and informally, for long terms or temporarily, around common problems in order to derive a solution. These actor networks behave as though they are autonomous organizations and they can cross boundaries of the formal hierarchical structures of organizations and industries. Nevertheless, actor networks influence and are simultaneously influenced by the larger network of organizations, industries, and professions that contextually surrounds them. Actors commonly involved in standards making include physicians, nurses, technicians, pharmacists, therapists, social workers, government agencies, professional associations, independent consultants and researchers [5, 35].

In the actor network perspective, the standards making process is viewed as a process of balancing possible differences among actor views to create a single long-term

solution to recording, storing, and sharing clinical information [17]. However, the heterogeneity of the actors involved in standards making is posited as a barrier to achieving stable closed standards. The rationale is that the multiplicity of human actor perspectives, intentions, constraints, challenges, and agendas interact with and reorganize one another. In the process they tend to reinforce existing order or create disorder. Existing order is reinforced because participants in SDOs are strongly influenced by their perceived view of practice in their own domain including *a priori* relationships, terminology, practices, and artifacts [35]. Protocols tend to evolve to reflect the historical relationship structure of practitioners in the domain, the incentives that favour certain practices and discourage other practices, and the distribution of resources such as specialized medical equipment. However, disorder is injected into the network when, to adopt a standardized EMR system, physicians need to adapt their current processes to the processes embedded in the EMR system. An EMR embeds specific classification schemes and terminologies, and enforces standard means of conducting clinical procedures and evaluating performance [17]. These embedded standards are intended to specify, routinize, and make uniform the type and format of clinical information to be collected. As such, standards intervene in the learned practices, goals and localized responses that physicians have been enacting [35]. In this case, attempts to enforce standard protocols on care redirect the courses of patients, instruments, drugs, and staff.

The foregoing is not to say that forcing change, even if it is uncomfortable for actors, is not a valid strategy. However, the trajectory of projects (i.e., the trend toward or away from desired objectives), such as the integration of information in EMRs, depends on local networks of actors to mobilize (i.e., achieve) the steps of projects [18]. In healthcare, local actors such as physicians have the preponderance of power and can simply choose not to cooperate, or to sabotage the protocol by not entering patient information [35]. Physicians may perceive a project as a threat, to the extent that standardization is perceived as a process of centralizing control over medical practitioners [17]. Indeed, standards have been viewed as “tyrannical domination” mechanisms and “total control of physicians' doings and non-doings” [35, p. 287, 291]. Instead, healthcare professionals treat standards as equivocal suggested guidelines rather than protocols with which they should automatically comply. Thus, overly restrictive standards can lead to a divergence of actor trajectories rather than bringing them together as a universal standard. Divergence is manifest in the cycles of specifying and renegotiating standards as unintended side effects reflect back on the standards making process. Thus, reflexivity also amplifies social complexity in standards making.

In the social arena, reflexivity refers to the observation that changes to standards by one group of actors have unintended side-effects that change the nature of the actor-networks in which the standards makers are immersed, which reflects back to influence the standards making process, and so on [17]. For example, when a standard is implemented it can unintentionally affect the interests at stake, the distribution of costs, the careers involved, the technologies which are selected, and which diagnostic tests are deemed more crucial than others [35]. Not until that moment do the standards makers see the scope and depth of the standard's impact on pre-existing practices, patients' lives, histories and futures that are not explicated in the standard. Attempts to manage these effects become increasingly difficult as the installed base of the standard grows over time [17].

We have shown that scholars of technical and social complexity perspectives share one conclusion: assumptions of universality in standards lead to failure. Standards makers seek order through a reductionist process of breaking down and explicating the medical domain under the assumption that they can achieve a universal standard beneficial to medical practice [17]. These explicated standards are then embedded in healthcare information systems such as EMRs. The approach is well grounded in engineering principles of consistency and non-redundancy. However, such reductionist techniques cannot be generalized to all localized problems. The assumptions, and the restrictive standards that result from them, can become a risk when the underlying assumptions fail. This finding is better understood in light of the important roles that different levels of situational equivocality and different sources of uncertainty play in how actors evaluate standards in practice, as discussed next.

4 Perceived Value of Information: Uncertainty vs. Equivocality as Factors

As discussed previously, one driver of standards making is actors' desires to overcome the uncertainty that they experience when they do not have information essential to inform care decisions. Daft and Lengel [10] find that, in addition to uncertainty, equivocality plays a distinct but important role in perceived information needs of actors and, thus, their perceived value of the mechanisms for satisfying their information needs.

Equivocality refers to the presence of multiple, conflicting interpretations of conditions. Whereas the solution for uncertainty lies in providing a sufficient amount of highly structured information to actors, the solution to equivocality requires a sufficient richness of information to enable actors to make sense of problems that are not well structured [10, 15, 24]. Thus, actors in uncertain situations may value standards that structure the problem and make it easier for actors to find information in structured records. However, actors in equivocal situations cannot structure the problem to enable them to find answers in structured records and require, instead, rich mediums such as person-to-person interaction to overcome equivocality. In equivocal situations, standards restrict users and interfere with their need to overcome equivocality. Equivocality also arises when interacting actors are highly differentiated, such as a medical lab versus a general practitioner [15]. Different values, perspectives and frames of reference lead to conflicting interpretations of information (i.e., equivocality) recorded in one context and interpreted in another. Therefore, the perceived value of information exchanged via EMRs diminishes as the differences between sources and users increase.

Furthermore, in situations involving uncertainty, the source of the uncertainty determines the perceived value of standards to the user. When uncertainty arises from interdependencies between subunits (e.g., a hospital physician relies on the x-ray department for diagnostic images), data standards expedite communication by providing a standardized, formalized language with which to communicate between subunits [15, 38]. Conversely, when uncertainty arises from task complexity or environmental instability, mandatory compliance with inflexible standards restricts the ability of actors to adapt to local conditions and find workable solutions. Therefore,

standardization tends to improve the speed and efficiency of information flows between like units but at the cost of flexibility and innovation. Closed standards imposed on like actors leads to better communications and better coordination; whereas closed standards imposed on highly differentiated actors leads to more compromise, more design costs, and more bureaucratic delay.

We can, however, make some recommendations to mitigate the paradoxical cycle. The proposed solutions involve a different conceptualization of standards and the standards making process, which suggest new directions for the research and practices of making and embedding standards in healthcare information systems.

5 Recommendations

Given, as argued in the preceding analysis, that assumptions of universality fail when trying to develop standards, it is clear that we need a more flexible conceptualization. For example, Hanseth *et al.* [17, p. 577] state that “we need to accept in our standard making that our complex worlds are populated with a multiplicity of orders that are inconsistent (and) we need to be able to live with such multiplicities and inconsistencies”. To that end, we make the following recommendations for future research and practice.

Recommendation 1. The first recommendation is to uncouple relatively stable technical components from social issues. This may make more technical problems solvable [17]. Standards makers can uncouple standards from contentious debates over “appropriate” protocols by avoiding the embedding of specific working practices into the standards. Different values, perspectives, and frames of reference make it more difficult to get actors to agree on protocols that alter their own practices than it is to obtain agreement on standard definitions for data items and events [15].

Leaving actors some leeway or discretion is sometimes necessary to garner their cooperation and mitigate the reflexive cycle [35]. Therefore, Timmermans and Berg [35] recommend that we subordinate specification of standards that embed protocols to the practitioners. Protocols that evolve from practice instead have the support of practitioners because they address the real problem that practitioners face. By delegating the task of maintaining and producing the protocol's requirements to medical personnel, these researchers find that not only can a standardized protocol be achieved, but the very process also becomes a stabilizing factor in standards development. They observed that actors subtly but importantly remind each other of the protocol in daily practice while evolving the protocol to incorporate ad hoc processes necessary to address unforeseen contingencies. Protocols are made universal when practice is sufficiently crystallized. Thus, the process is iterative and, while it may not result in a standard as originally conceptualized by standards makers, it can produce beneficial results [17].

By uncoupling some technical problems from social issues we can identify elements of medical practices, instrumentations, and terminologies that are relatively stable. For example, highly standardized and integrated information systems are most perceived as useful where subunits are very interdependent and most easily implemented where sub-units are not highly differentiated. Stable elements in highly

interdependent and similar environments are, thus, the appropriate candidates to be explicated, ordered, and turned into standards [17].

Recommendation 2. Where information essential to care decisions cannot be reduced to explicit data items and pre-defined protocols in advance, we should re-conceptualize the role of standards as specifying a domain at a more abstract level. For example, Timmermans and Berg [35] advocate conceptualizing standards as a coordinating tool rather than attempting to specify with all detail every aspect of a domain. An example in practice is the HL7 Clinical Document Architecture (CDA). The CDA embodies a very different approach to standards making. Rather than specifying each detail of data for exchange, CDA was designed as a freestanding content document [1, 20]. In effect, CDA is an implementation of the Minimum Data Set (MDS) concept. MDSs have been developed for a variety of healthcare disciplines and organizations. Another example of MDS is the Continuity of Care Document (CCD), a standard for minimal data set transmission that builds on the HL7 CDA and has been under collaborative development by HIMSS EHRVA (Health Information Management and Systems Society Electronic Health Record Vendors Association) [11]. It can include a core data set that enables the most common facts about patient healthcare to be recorded by one healthcare professional along with free-form content that conveys contextual information essential for another practitioner to interpret the record [17]. This allows the interchange of essential information between EMR systems without the need to modify their databases with numerous fields intended to represent the same information. It thus attempts a balance between standardization and flexibility.

The preceding discussions begin to elucidate how the standards making process affects physician perceptions of the value versus risk of adopting standardized healthcare information systems. It is necessary to understand the barriers to adoption before solutions can be purposefully developed. However, the recommendations begin to define at an abstract level where potential lies for improving standards making and physicians' perceived value of information systems that embed standards. These recommendations need to be defined further and substantiated in future research of healthcare standards.

6 Conclusions

This study focuses on the need to improve information flows among healthcare practitioners in order to increase the perceived value of integrated healthcare information systems among physicians. Data exchange among healthcare systems fails when different healthcare practitioners use different definitions, terms, identifiers, file formats, and rules to record information [38]. Therefore, the healthcare industry has invested substantially in embedding predefined standards into healthcare information systems with the intention that information recorded by one healthcare practitioner can be accessed and interpreted by another with consistency [17]. However, deeper investigations illustrate that the standards making process tends to reflect and reinforce the practices of some actors while entering a self-destructive spiral of unintended consequences, re-specification, and more unintended consequences when implemented by others. Furthermore, the sheer complexity of standards making in healthcare can

explain much of the failure of standards being used in practice. Standards making processes that employ a solely reductionist approach fail because the implicit assumption that medical practice can be universally defined in data and protocols explicated in advance does not hold up in practice. Instead, standards present restrictive barriers to practitioners when they face unforeseen contingencies or equivocal situations, and exchange information across different environments. These findings offer a compelling explanation for at least part of the low adoption rates among physicians.

We have articulated two conceptual directions for future research and changes to standards making practice that we believe would assist in moving towards achieving better standards and interoperability solutions. By uncoupling stable components from contentious protocol issues and by re-conceptualizing the role of standards to embrace greater flexibility in some areas, we believe standards makers can play a significant role in increasing the perceived value of standards and the information systems that embed the standards.

References

1. Alschuler, L., Beebe, C., Boyer, S., Dolin, R.H.: Clinical Document Architecture Framework Release 2.0, <http://www.hl7.org/Library/Committees/structure/CDA%5FFramework%2E1%2E01%2E4%5F08%5F02%2Edoc> (accessed February 12, 2009)
2. Alvarez, R.: Accelerating the Development and Implementation of Electronic Health Records (EHR) in Canada, <http://infranet.uwaterloo.ca/inftalks/2005-2006/2005-11-23/default.pdf> (accessed February 12, 2009)
3. Anderson, J.G., Balas, E.A.: Computerization of Primary Care in the United States. *International Journal of Healthcare Systems and Informatics* 1(3), 1–23 (2006)
4. Archer, N.: Mobile eHealth: Making the case. In: First European Mobile Government Conference, Brighton, England (2005)
5. Archer, N., Cocosila, M.: Improving EMR System Adoption in Canadian Medical Practice: A Research Model and Survey Proposal. McMaster eBusiness Research Centre Working Paper Series: Paper #22. DeGroote School of Business, McMaster University, Hamilton, Ontario, pp. 1–23 (2008)
6. Bergeron, F., Raymond, L.: Managing EDI for Corporate Advantage: A Longitudinal Study. *Information & Management* 31(6), 319–333 (1997)
7. Chernos, S.: Cross-country check-up. *Technology for Doctors*, <http://www.canhealth.com/D07oct.html> (accessed February 12, 2009)
8. Cilliers, P.: *Complexity and Postmodernism: Understanding Complex Systems*. Routledge, London (1998)
9. Clarke, D., Howells, J., Wellingham, J., Gribben, B.: Integrating Healthcare: the Counties Manukau Experience. *Journal of the New Zealand Medical Association* 116, 1169 (2003)
10. Daft, R.L., Lengel, R.H.: Organizational Information Requirements, Media Richness, and Structural Design. *Management Science* 32(5), 554–571 (1986)
11. Electronic Health Record Vendors Association (EHRVA)., *Quick Start Guide HL7 Implementation Guide: CDA Release 2 – Continuity of Care Document (CCD)*. Healthcare Information & Management Systems Society (HIMSS) (2007)

12. European Commission: Citizen's Summary: Better Health Treatment for Travellers and Expats in the EU. European Commission: Brussels, Belgium, http://ec.europa.eu/information_society/activities/health/docs/policy/20080702-interop_recom_citizen_summary.pdf (accessed February 12, 2009)
13. European Union: Commission Recommendation on Cross-border Interoperability of Electronic Health Record Systems. Commission of European Communities: Brussels, Belgium, http://ec.europa.eu/information_society/activities/health/docs/policy/20080702-interop_recom.pdf (accessed February 12, 2009)
14. Goodhue, D.L., Quillard, J.A., Rockart, J.F.: Managing the Data Resource: A Contingency Perspective. *MIS Quarterly* 12(3), 373–392 (1988)
15. Goodhue, D.L., Wybo, M.D., Kirsch, L.A.: The Impact of Data Integration on the Costs and Benefits of Information Systems. *MIS Quarterly* 16(3), 292–311 (1992)
16. Grimson, J., Grimson, W., Hasselbring, W.: The SI Challenge in Healthcare. *Communications of the ACM* 43(6), 49–55 (2000)
17. Hanseth, O., Jacucci, E., Grisot, M., Aanestad, M.: Reflexive Standardization: Side Effects and Complexity in Standard Making. *MIS Quarterly* 30, 563–581 (2006)
18. Heeks, R., Stanforth, C.: Understanding e-government project trajectories from an actor-network perspective. *European Journal of Information Systems* 16(2), 165–177 (2007)
19. HIMSS, Interoperability Definition and Background, http://www.himss.org/content/files/interoperability_definition_background_060905.pdf (accessed February 12, 2009)
20. Jagannathan, V.: Enterprise Integration Approaches in Healthcare: A Decade of Trial and Error. In: Bernus, K.P., Fox, M. (eds.) *Knowledge Sharing in the Integrated Enterprise: Interoperability Strategies for the Enterprise Architect*, IFIP, pp. 315–324. Springer, Heidelberg (2005)
21. Keshavjee, K., Bosomworth, J., Copen, J., Lai, J., Kucukyazici, B., Lilani, R., Holbrook, A.: Best Practices in EMR Implementation: A Systematic Review. In: 11th International Symposium on Health Information Management Research, Dalhousie University, Halifax, Nova Scotia, pp. 1–15 (2006)
22. Khoubati, K., Themistocleous, M., Irani, Z.: Evaluating Integration Approaches Benefits Adopted by Healthcare Organizations. In: *Twelfth European Conference on Information Systems*. Turku School of Economics and Business Administration, Turku (2004)
23. Khoubati, K., Themistocleous, M., Irani, Z.: Evaluating the Adoption of Enterprise Application Integration in Healthcare Organizations. *Journal of Management Information Systems* 22(4), 69–108 (2006)
24. Klein, G., Moon, B., Hoffman, R.F.: Making Sense of Sensemaking I: Alternative Perspectives. *IEEE Intelligent Systems* 21(4), 70–73 (2006)
25. Kolodner, R.M., Cohn, S.P., Friedman, C.P.: *Health Information Technology: Strategic Initiatives, Real Progress in Health Affairs*, Washington, DC, <http://content.healthaffairs.org/cgi/content/abstract/hlthaff.f.27.5.w391v1> (accessed February 12, 2009)
26. Lee, J., Cain, C., Young, S., Chockley, N., Burstin, H.: The Adoption Gap: Health Information Technology in Small Physician Practices. *Health Affairs* 24, 1364–1366 (2005)
27. Lee, S., Leifer, R.P.: A Framework for Linking the Structure of Information Systems with Organizational Requirements for Information Sharing. *Journal of Management Information Systems* 8(4), 27–44 (1992)

28. National Library of Medicine: SNOMED CT Affiliate License Agreement. U.S. National Library of Medicine: Bethesda, MD, <http://www.nlm.nih.gov/research/umls/metaa2.html> (accessed February 12, 2009)
29. Nøhr, C., Boye, N.: Towards Computer Supported Clinical Activity: A Roadmap Based on Empirical Knowledge and some Theoretical Reflections. In: Kushniruk, A.W., Borycki, E.M. (eds.) *Human, Social, and Organizational Aspects of Health Information Systems*, pp. 67–83 (2008)
30. OntarioMD: Clinical Management Systems Specification Appendix B – Data Portability Requirements (October 24, 2008). Version: 3.01. OntarioMD Inc., <https://www.ontariomd.ca/portal/server.pt?space=CommunityPage&control=SetCommunity&CommunityID=478&PageID=1591#spec> (accessed February 12, 2009)
31. Picard, A.: *For Health's Sake, Trash Those Paper Records*. Globe and Mail, Toronto (2007)
32. POSP, In The Know. In: Collins, L. (ed.) *POSP Newsletter*, Alberta NetCare, Edmonton, Alberta, pp. 1–6 (May 2006)
33. Schneberger, S.L., McLean, E.R.: The Complexity Cross: Implications for Practice. *Communications of the ACM* 46(9), 216–225 (2003)
34. SSHA: ONE Network. Smart Systems for Health Agency: Toronto, Ontario, http://www.ssha.on.ca/products-services/one_network.asp (accessed February 12, 2009)
35. Timmermans, S., Berg, M.: Standardization in Action: Achieving Local Universality Through Medical Protocols. *Social Studies of Science* 27(2), 273–305 (1997)
36. Urowitz, S., Wiljer, D., Apatu, E., Eysenbach, G., DeLenardo, C., Harth, T., Pai, H., Leonard, K.: Is Canada Ready for Patient Accessible Electronic Health Records? *A National Health Scan. BMC Medical Informatics and Decision Making* 8(1), 1–33 (2008)
37. Wixom, B.H., Watson, H.J.: An Empirical Investigation of the Factors Affecting Data Warehousing Success. *MIS Quarterly* 25(1), 17–41 (2001)
38. Wybo, M.D., Goodhue, D.L.: Using Interdependence as a Predictor of Data Standards: Theoretical and Measurement Issues. *Information & Management* 29(6), 317–330 (1995)
39. Yendt, M.: Overview of the MARC III Project Including Opportunities for the IT Community. In: *eHealth Technology Showcase*, November 18. Mohawk Applied Research Centre, Hamilton (2008)
40. Yendt, M., Bender, D., Minaji, B.: Developing an Open Source Reference Implementation of the Canadian Electronic Health Records Solution. *Open Source Business Resource* (November 2008), <http://www.osbr.ca/ojs/index.php/osbr/article/view/776/747> (accessed February 12, 2009)

Aligning Goal and Value Models for Information System Design

Ananda Edirisuriya and Jelena Zdravkovic

Department of Computer and Systems Sciences
Stockholm University and Royal Institute of Technology
Forum 100, SE-164 40 Kista, Sweden
{si-ana, jelenaz}@dsv.su.se

Abstract. The success of process-aware information systems and web services heavily depends on their ability to work as catalysts for the business values that are being exchanged in a business model. The motivation of a business model can be found in the goals of an enterprise which are made explicit in a goal model. From the IT perspective, goal and business models form part of a chain of models, ending with an information system model. Thereby, analyzing and establishing the alignment of business models with goal models is a starting task on the way to a business-aware information system. This paper discusses the alignment of value-based business models with system-oriented goal models. The result is a set of transformation rules between the two models. A case study from the health sector is used to argument the way we ground and apply our contribution.

Keywords: Goal Modelling, Business Modelling, Alignment.

1 Introduction

Requirements Engineering (RE) is the process of deriving, validating and maintaining systems requirements [1]. Goal and value perspectives are two views used in developing such requirements. In Goal-Oriented Requirements Engineering (GORE), strategic objectives are exploited as the basis for the RE process [2]. Value-oriented requirements engineering explores the concept of economic value during the requirements engineering process [3].

Different models are used to capture these two views. The strategic goals are captured in goal models, and economic values are considered in business models. Goal models represent interests, intentions, and strategies of different actors. Goal models can be employed as a driving force for eliciting business activities and alternative ways of doing them, giving thereby a motivation for ‘why’ certain decisions are pursued. Business models identify actors, economic resources (i.e. *values*), value transfers among actors, and how values are created and offered. Business models therefore give a high-level view by focusing on the ‘what’ of a business.

The business model should be aligned with the strategic goals of the enterprise. That is, the realisation and the characteristics of the business model should be in

accordance with the enterprise's long-term interests. Otherwise, values being offered and exchanged are not motivated properly, neither involvement of certain business actors. In this situation, even if the system model is derived from the business model it would not be aligned with the intentional dimension of the business. Thereby, in the RE process there is a need for structured methods to align enterprises' business models with the interests of stakeholders.

In this paper, we investigate the relation between goal models and business models. To address the described problem, we attempt to explain goal model components using business model components. The paper contributes to the research question: *How can a goal (intentional) model and a business model be aligned with each other?*

The paper is structured in the following way. In Section 2, we discuss the related research. In Section 3 and Section 4, we describe the goal modelling and business modelling language we used in this work. In Section 5, we propose the transformations required to move from one model to another. In Section 6, we apply our method to a case study from the Swedish health sector. The paper concludes with a discussion of the results and the possibilities for future research in Section 7.

2 Related Works

A number of structured methods and tools have emerged over the last two decades to facilitate the task of designing business models. The business model ontologies provide a constructive basis for the design of business models [4]. In this direction, a number of research works have focused on the task of designing business modelling ontologies [3][5][6]. These ontologies contain concepts to be included in business models, their relationships and constraints.

The Resource-Event-Agent (REA) ontology was originally developed to model the change of values of accounting information systems in an organization [5]. It has since been extended and used in e-commerce frameworks [7] and enterprises' information systems architectures [8]. The *e³value* business ontology has been developed with the intention of modelling value networks of cooperating business partners [3]. It provides a set of tools to analyse and determine the sustainability of value networks. In addition to model transfer of resources, Business Modelling Ontology (BMO) encompasses internal capabilities of enterprises, resource planning, value propositions and marketing aspects [6].

There are other researches that concerns aligning business models with organizational goals. In [9], relationships between goals and business models are investigated in the context of cross-organizational environments, where networks of enterprises jointly satisfy consumer needs. The approach involves a requirements analysis based on cross-organizational goals and values. The analysis helps in understanding how value modelling can be used in finding detailed goals of enterprises and conflicts among those goals. A method for transforming i* goal modelling language [10] to the *e³value* model is discussed by the authors in [11], with the aim of exploring commercial e-services from a strategic and profitability perspective. The method starts with modelling goal, task, resource and soft goal dependencies among collaborating actors and ends with deriving a goal-aligned business model. A set of guidelines are proposed to map the elements of the two models. In [12], the authors discuss a method to

bridge the gap between the Business Motivation Model [13] and the e^3 value model [3]. The work suggests a way to formulate goals in terms of elements in the business model. The bridging is done by use of a set of means templates. Each means describes a way to achieve the goals. The means are expressed using business model notions. The method starts with an existing e^3 value model and derives a new e^3 value model that conforms to the goals using some production rules.

In this work, we employ a goal modelling language used for deriving systems requirements, to model requirements of value networks. We propose a way to formulate the elements of goal models in terms of business model concepts. A set of templates are proposed to capture requirements in value networks. Finally, a set of guidelines are established to construct a complete e^3 value model from the goal model.

3 Goal Modeling

Goal models are used to capture intentions (i.e. conditions) of enterprise stakeholders. They provide business motivations and directions towards concrete actions. Goal modelling is being used in numerous engineering contexts such as business analysis [13], requirements engineering for system development [15] and organizational-oriented modelling [10].

The purpose of this work is to analyse and design goal-aligned business models in the context of information system design. Therefore, we will use KAOS [15], a well-established goal-modelling language in system requirement engineering. We continue this section by introducing a business case study, which is further used for describing KAOS goal-modelling language.

3.1 Case Study: Eye-Care Health Service

We explain here our case study from the Swedish eye-care domain. The case encompasses three actors, Patient, Primary Care Physician and Eye-Care Specialist. When a patient needs an eye treatment, he goes to a primary care physician. After diagnosis the primary care physician provides a treatment to the patient, getting a fee in return. If more advanced treatments are necessary, the patient will be referred to an eye-care specialist. In this case, the referral information will be sent to the eye-care specialist by the primary care physician. The eye-care specialist provides an advanced treatment to the patient and gets a fee in return. A more detailed description of the case can be found in [16].

3.2 The KAOS Goal Modeling Language

The Keep All Objects Satisfied (KAOS) language is used in goal-oriented requirements engineering to collect system requirements using organizational goals. The KAOS was extended in [17] to model various dependencies of the agents involved. We briefly discuss here the concepts of KAOS used in this study. Fig. 1 shows an example of a KAOS goal model.

The main concepts of KAOS are *goals*, *objects*, *agents* and *operations*. A *goal* in KAOS is described as a desired system property specified by a stakeholder. A system property could be a service or quality of a service offered by the system. An *object* is

a ‘thing’ that can be distinctly identified in a goal statement. An object can be an entity, event, agent or relationship. An *entity* is an autonomous object, such as ‘Treatment’. An *event* is an instantaneous object that represents a certain incident occurring at some point in time. For example, ‘Request for an appointment with Eye-Care Specialist’. An *agent* is an active object, who is capable of performing some activities. An agent can be an organization, human, device or software. A *relationship* is a connection between some objects. For example, an intended relationship between a ‘Patient’ and an ‘Eye-Care Specialist’. An *operation* is an action performed by an agent in the environment. The operations can have pre-, post- and trigger conditions. An event could be used to trigger or stop the execution of an operation..

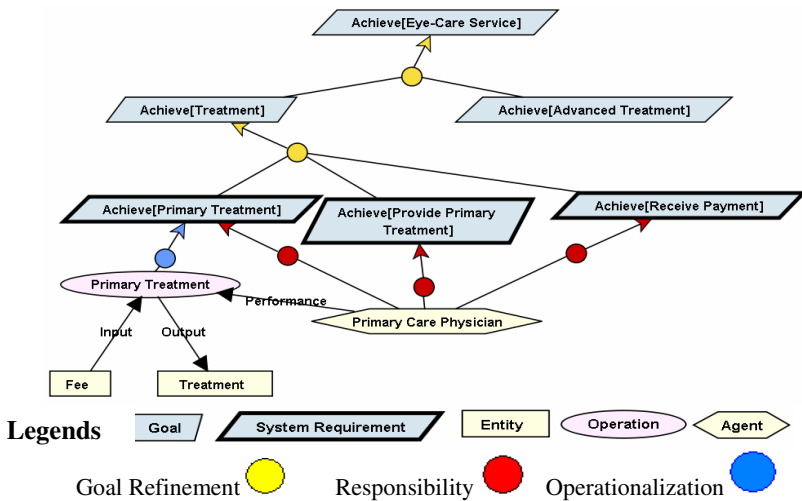


Fig. 1. An example of a KAOS goal model

In addition to the above constructs, KAOS defines certain relationships between these constructs. The high-level goals can refine into low-level systems requirements by means of a *refinement* relation. A *system requirement* is a low-level goal that can be placed under the responsibility of a single agent. For instance, in Fig. 1, the high level goal *Achieve[Eye-Care Service]* is refined into sub-goals *Achieve[Treatment]* and *Achieve[Advanced Treatment]*. The former is refined into system requirements *Achieve[Primary Treatment]*, *Achieve[Provide Primary Treatment]* and *Achieve[Receive Payment]*. A *responsibility* relationship is used to assign a system requirement to an agent. The system requirement *Achieve[Primary Treatment]* is assigned to the agent ‘Primary Care Physician’. An *operationalization* relationship is used to indicate the operation that realized the system requirement. The *performance* relation describes the operation that realized the system requirement. The *input* and *output* links describe the objects that are being utilized and produced by an operation. In our example, the operation ‘Primary Treatment’ is performed by the agent ‘Primary Care Physician’ and it takes the object ‘Fee’ as the input and produces the object ‘Treatment’ as the output.

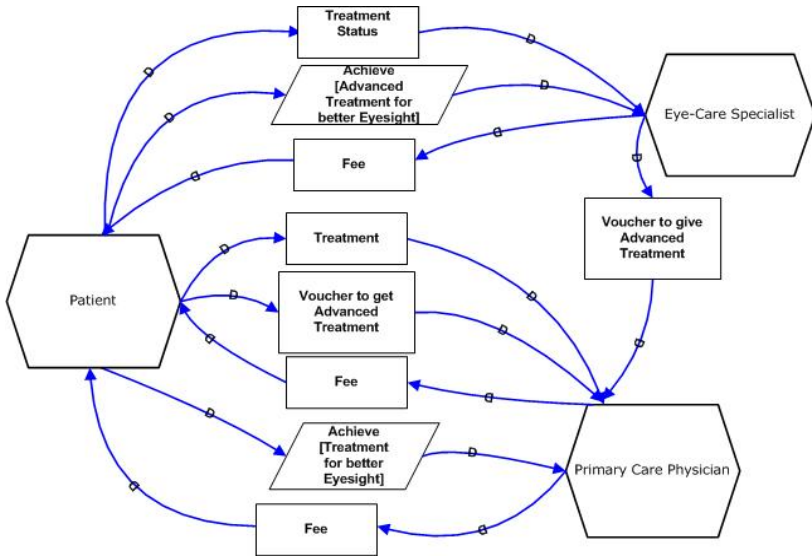


Fig. 2. Agent dependency diagram for the health care case

To complement the basic goal model, KAOS also provides some additional models, such as *object*, *agent responsibility*, *operation*, *agent interaction* and *agent dependency*. To describe our work we only need the agent dependency model. The KAOS uses *i** notion of dependency to describe agent dependencies [17]. A depender agent depends on a depensee agent on a certain *dependum* to be satisfied. The dependum can be a goal, soft-goal, operation, or entity. Fig. 2 shows an agent dependency diagram for the described case study. A ‘Patient’ depends on an ‘Eye-Care Specialist’ for the goal Achieve[Advanced Treatment for better Eyesight]. The ‘Primary Care Physician’ depends on ‘Patient’ for the entity (type of object) ‘Fee’. In Fig. 2, we represent a goal by an angle rectangle, entity with a rectangle and an agent by a hexagon.

3.3 The KAOS Requirements Engineering Process

We briefly explain here the KAOS requirements engineering process. The details can be found in [15]. First, a goal model is constructed. The high-level goals are refined into sub-goals until each sub-goal cannot be further refined and can be kept under the responsibility of a single agent. These low-level goals, also known as system requirements, drive the identification of objects and operations. Once the objects have been identified system requirements can be assigned to agents. An operation may use input objects and produce output objects. These operations are responsible for realizing goals.

4 Business Modeling

Enterprises engage in activities that create economic values, which are then exchanged with business partners and customers. A business model of an enterprise

describes what values an enterprise offers to its customers and the architecture of the enterprise with its network of partners for creating, and delivering, those values.

In the literature there are a number of languages and ontologies for modelling enterprise businesses. Among them, the three widely used ontologies are REA [5], e^3 value [3] and BMO [6]. In this study we use e^3 value ontology as our business modelling language as it is easy to understand and is the most suitable for modelling and visualizing value networks (i.e. multi-actor business collaborations). In the next section, we briefly explain the main concepts of the e^3 value ontology.

The e^3 value Business Model

The e^3 value business ontology focuses on modelling value networks of cooperating business partners [3]. The ontology models exchanges of the objects of economic value (*value objects*) between actors involved in business collaborations. An example of an e^3 value model is shown in Fig. 3.

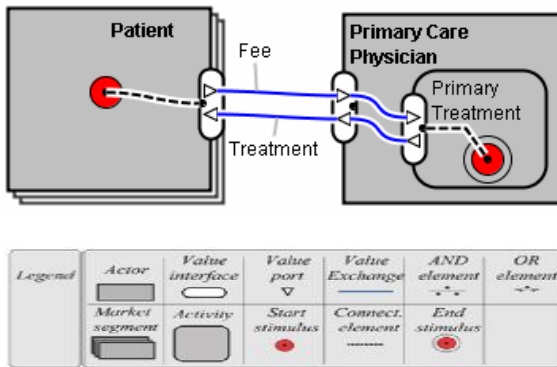


Fig. 3. e^3 value model for the health care case

The main concepts used in e^3 value ontology are: *actor*, *value object*, *value port*, *value interface*, *value exchange*, *value activity* and *market segment*. An *actor* is an economically independent entity capable of performing activities that create values and transfer values. An actor must be able to take economic decisions and responsible for profit-and-losses. ‘Eye-Care Specialist’ and ‘Ministry of Health’ are examples of actors. A *value object* is something that one actor transfers to another actor. A value object has an economic value for at least one of the actors involved in the transfer. ‘Treatment’ and ‘Fee’ are examples of value objects. A treatment is of value for the patient as it can improve the eyesight of the patient. A *value port* is used to provide or receive value objects to or from other actors. A value port has a direction, *in* or *out*. An *in-port* is used to receive a value object (e.g. receive a fee) and an *out-port* is used to provide a value object (e.g. provide a treatment). A *value interface* groups value ports and consists of in-port(s) and out-port(s) that belong to the same actor. A *value exchange* is a pair of value ports of opposite directions belonging to different actors. A value exchange represents a potential value object trade. In a business scenario, when an actor gives up something, s/he expects something in return. This is known as

economic reciprocity. For example, ‘Primary Care Physician’ provides ‘Treatment’ for ‘Patient’ and gets a ‘Fee’ in return. In e^3 value economic reciprocity is modeled through *value interfaces*. A *value activity* is an operation that produces value objects and could be carried out in an economically profitable way for at least one actor. The ‘Primary Care Physician’ uses the value activity ‘Primary Treatment’ to treat a patient. For this purpose it requires the ‘Fee’ at its value in-port. A *market segment* is a group of value interfaces belonging to actors, who may value exchanging economic objects equally. A ‘Patient’ is a market segment. Fig. 3 shows that a ‘Patient’ obtains the ‘Treatment’ from ‘Primary Care Physician’. Actors within the ‘Patient’ market segment equally value the object ‘Treatment’.

5 Methodology for Model Transformation

The KAOS is a *goal-oriented* requirements elaboration method used to derive requirements for a future system from high-level goals. The e^3 value ontology is developed with the purpose of modelling value networks of cooperating business partners. We focus in this section on how to construct an e^3 value model starting with a KAOS goal model.

5.1 Goal Formulation

Formulation of goals is a difficult task, as argued by authors in [12]; goals can vary from day-to-day operational goals to strategic goals of enterprises. To overcome this problem we suggest how to formulate goals in a uniform way using business model components. KAOS provides a set of goal patterns to be used as guidelines for acquisition and definition of goals. These patterns are based on the temporal behaviour required by the goals. KAOS identifies four such goal patterns; Achieve, Cease, Maintain and Avoid [15].

We formulate a goal as a *desired condition on one or more features (property) of a resource (entity)*. The patterns used in KAOS are extended with the introduction of temporal behaviour to this goal expression.

Achieve goal: This goal requires a condition on one or more features of a resource to hold.

Cease goal: This goal requires a condition on one or more features of a resource to stop holding.

Maintain goal: This goal requires a condition on one or more features of a resource to hold always.

Avoid goal: This goal requires a condition on one or more features of a resource never to hold.

For example, in the eye-care case, goal ‘Treatment Obtained’ can be formulated as Achieve[Treatment (resource) for better (condition) eyesight (feature)]. The objective of the eye treatment (resource) is to have better eyesight (feature) at some point in time. The modeller finds it is easy to formulate goals in this way and makes goal expressions more uniform.

5.2 System Requirements Templates

Goal models can be constructed using the goal formulation described above. Once a goal model is constructed, high-level goals should be refined to sub-goals and further to the level of system requirements.

Here, a question arises - how can these system requirements be modelled? In the context of organizational business, where actors exchange resources and money, most of the actions are about acquisition, production, or provision of resources. The authors of this paper are of the opinion that the requirements of value networks can be modeled using these business model notions. In other words, requirements can be expressed using the basic entities of business models describing the actors, resources, transfer of resources, and the activities required to produce and consume these resources. Thus, we can use a small number of templates to formulate requirements involved in modelling value networks of cooperating business partners. A template design so has three parts: an operation, a resource and an agent, i.e. *Template: [Operation, Resource, Agent]*.

We propose the following set of templates to construct system requirements.

1. Provide (Operation) Resource (Entity) to Agent
2. Receive (Operation) Resource (Entity) from Agent
3. Produce [Use, Consume] (Operation) Resource by Agent

5.3 Transforming KAOS to the e^3 value Ontology

In this section we explore the transformation steps between KAOS and e^3 value business model ontology. In a summary, the KAOS goal model complements the e^3 value model by exposing the strategic reasoning behind its value creating and value transferring activities. The notions of goals, system requirements, objects, responsibility assignments, operations and agent dependencies of the KAOS model provide a rich set of concepts to model the range of interests of actors involved in a value network.

We define the following detailed guidelines for constructing an e^3 value model from KAOS. Let D be an agent dependency model in KAOS.

1. For each *agent* in D , corresponding legal agent becomes an *actor* or *market segment* in e^3 value depending on whether the agent involves is a single agent or multiple agents (who assign same value to objects).

An agent in KAOS is an active object who is able to perform some manual or automated operation. An actor or market segment in e^3 value model is an economically independent entity who is capable of performing some activities to create and transfer values. Further, e^3 value model actors and market segments are responsible for taking economic decisions and profit and losses. Thus, a legal agent who is responsible for a KAOS agent can be mapped to actor or market segment in e^3 value model. For example, if KAOS agent is a coffee machine then the company who owns the coffee machine is becomes the actor in e^3 value model.

2. A *resource* (entity) R , involves in a *dependency* in D , becomes a *value object* in e^3 value model, if R has an economic value for at least one of the agent involved in the dependency.

An entity in KAOS is any autonomous object. This encompasses a broad class, from informational, tangible to non-tangible resources. Value objects in e^3 value model are products, money, services and consumer experiences, which actors think as valuable for them. Thus, only entities with economic values become value objects in e^3 value model.

3. A *dependency* in D becomes a *value exchange* in e^3 value model, if the corresponding entity involved in the dependency becomes a value object (guideline 2) in e^3 value model.

As explained in Section 3.2, in KAOS, a depender agent depends on dependee agent on a certain dependum to be satisfied. The dependum can be a goal, a resource (entity), or an operation. This can be interpreted as a value exchange in e^3 value model, if the dependency involves an economic resource.

4. A group of *dependencies* between two agents in D can be mapped to *value interfaces* of corresponding actors (market segments) in e^3 value model, if the associated dependencies are mapped to value exchanges (guideline 3) in e^3 value model.

In KAOS, dependencies reflect relationships between agents. In e^3 value model, actors exchange value objects through value interfaces. Therefore, a set of dependencies between two agents can be translated into value interfaces of corresponding actors or market segments, if the dependencies are mapped to the value exchanges.

5. An *operation* performed by an agent that produces an entity (with an economic value) in KAOS becomes a *value activity* of the corresponding *actor* or *market segment* (guideline 1) in e^3 value model.

An operation in KAOS is an activity executed by an agent and operates on objects. An operation may use input objects and may produce output objects. A value activity in e^3 value model is an operation that produces value objects. Thus, operation in KAOS is a value activity of the relevant actor (or market segment), if it produces an entity with an economic value.

6. Two *operations* (performed by the same agent) link with *input* and *output* links become *value exchanges* between corresponding *value activities* (guideline 5) of the corresponding actor or market segment (guideline 1).

This guideline is for modelling value exchange between value interfaces of two value activities of the same actor (market segment). Consider the two operations $op1$ and $op2$ in KAOS that are performed by the same agent. If output of $op1$ is an input in $op2$ and $op1$ and $op2$ become value activities in e^3 value model, then input and output relations of $op1$ and $op2$ become a value exchange between the two value activities.

7. If an *entity* (with an economic value) involved in a certain *dependency* in D is an input or output of a certain KAOS operation then there will be a *value exchange* between value interface of the corresponding value activity and value interface (introduced by guideline 4) of the corresponding actor or market segment.

This guideline is for modelling value exchange between a value interface of a value activity and value interface of the corresponding actor (market segment). An agent dependency might involve an entity and this entity might be an input or output of a certain KAOS operation. The operation is mapped to a value activity and agent dependency is mapped to a value exchange. Therefore, we can draw a value exchange between the corresponding value interfaces.

6 Method Application to Create a Goal-Based Business Model

This section discusses how business models can be aligned with goal models using the transformation guidelines defined in the previous section. The method involves taking an input goal model and constructing a business model that conforms to the goal model. The main tool used here is the template design for modelling requirements of a value network and transformation guidelines discussed in Section 5.

To apply this method, the goal modeller first needs to construct a goal model. In constructing the goal model, the modeller uses the goal patterns discussed in Section 5.1. To construct the requirements the modeller uses the templates discussed in Section 5.2. In addition, the modeller needs to construct the agent dependency model. Finally, using the guidelines discussed in Section 5.3, we can construct an e^3 value business model. The method can be summarized as follows:

1. The goal modeller constructs a goal model using the goal patterns and requirements templates.
2. The goal model will be extended by the addition of objects, agent responsibilities, and operations. Further, the goal model will be complemented with the agent dependency model.
3. The guidelines provided in Section 5.3 are applied to construct an e^3 value business model from scratch.

Method Application to the Running Example

In this section, we apply our method to the running example. First, the goal model is constructed from the goal patterns and requirements template from Section 5.

Fig. 4 shows an excerpt of a goal model designed for the eye-care case. The figure describes the top goals, sub-goals, system requirements, objects, operations and their relationships. Each system requirement in the goal tree is a leaf node. The goals are modelled using the patterns described in Section 5.1. For example, the goal Achieve[Primary Treatment (resource) for better Eyesight] is formulated using the Achieve goal pattern. This explains that the condition better eyesight (feature) holds for the resource 'Primary Treatment' at some point in time. To construct system requirements, templates given in Section 5.2 are used. For example, system requirement 'Provide Treatment to Patient' is constructed using template 1. Using the system requirements, the modeller can identify the objects, agents and operations. The goal model is complemented with an agent dependency model to show agent dependencies.

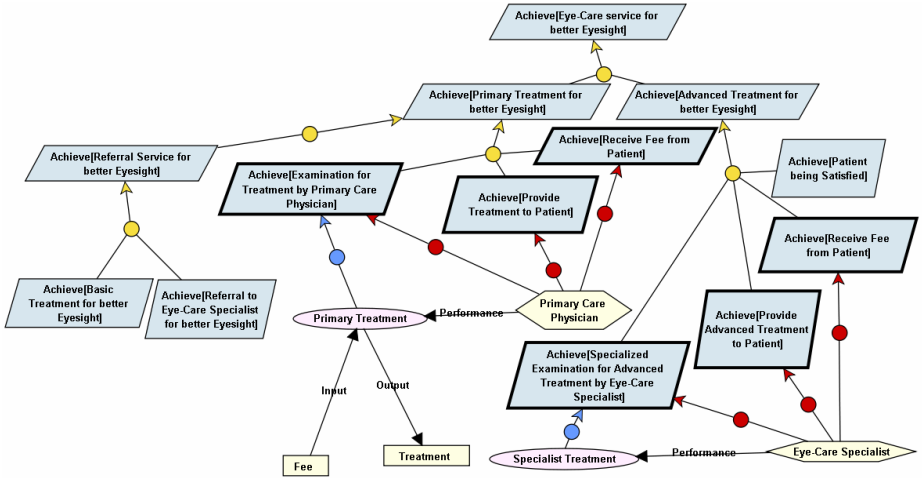


Fig. 4. Excerpt of KAOS goal model for the health care case

Identifying Objects

The objects can be identified using leaf nodes (system requirements) in the goal model. KAOS objects include entities, agents, events and relationships among these. Out of these constructs it is sufficient to identify entities and agents to construct business models. The events are more important in designing process models. To identify entities and agents we use template 1 and template 2. The objects identified from the goal model in Fig. 4 are shown in Table 1.

Table 1. Excerpt of objects relevant to the health care case

Object	Type
Patient	Agent
Primary Care Physician	Agent
Eye-Care Specialist	Agent
Fee	Entity
Treatment	Entity
Advanced Treatment	Entity
Voucher to get Advanced Treatment	Entity

Identifying Agent Responsibilities

Once the agents have been identified, the system requirements can be assigned to the agents. An agent is responsible for a particular requirement. For example, requirement ‘Specialized Examination for Advanced Treatment by Eye-Care Specialist’ is assigned to the agent ‘Eye-Care Specialist’.

Identifying Operations

The next step is to identify the operations, their inputs and output objects and assignment of operations to agents. For example, from the requirement ‘Examination (operation) for Treatment (resource) by Primary Care Physician (agent)’, we can identify the operation ‘Primary Treatment (Examination)’. This operation uses the input resource ‘Fee’ and produces output resource ‘Treatment’. It is the modeller’s responsibility to identify such information and show them in the goal model using input, output and performance links.

Construct e^3 value model

The guidelines provided in section 5.3 will help to construct the e^3 value model. The e^3 value model drawn for our running example is shown in Fig. 5.

Guideline 1: We identify three agents from our running example, ‘Patient’, ‘Primary Care Physician’ and ‘Eye-Care Specialist’. They are described as actors or market segments in e^3 value model (see ① in Fig. 5).

Guideline 2: In the agent dependency model, the goal dependency ‘Achieve[Treatment (resource) for better Eyesight]’ contains the resource (entity) ‘Treatment’, which has some economic value for the patient. Therefore, the entity ‘Treatment’ becomes a value object in e^3 value model (see ② in Fig. 5).

Guideline 3: The resource (entity) dependency ‘Fee’ from ‘Primary Care Physician’ to ‘Patient’ in the agent dependency model implies that former is depending on the latter for the entity ‘Fee’. This dependency maps to a value exchange in e^3 value model, as the entity ‘Fee’ is mapped to a value object. (see ③ in Fig. 5).

Guideline 4: Each group of dependencies between two agents is mapped to value interfaces. For example, two dependencies ‘Achieve[Treatment for better Eyesight]’ and ‘Fee’ between ‘Patient’ and ‘Primary Care Physician’ can be translated to value interfaces of ‘Patient’ and ‘Primary Care Physician’. This is because both dependencies have mapped to value exchanges in e^3 value model (see ④ in Fig. 5).

Guideline 5: The operation ‘Primary Treatment’ produces the entity ‘Treatment’, which has some value for ‘Patient’. Such an operation can be mapped to a value activity of ‘Patient’ in e^3 value model (see ⑤ in Fig. 5).

Guideline 6: We use an example to describe this guideline. Consider the two operations ‘Diagnose’ and ‘Special Treatment’ performed by the agent ‘Eye-Care Specialist’. The operation ‘Special Treatment’ uses the output (‘Result’) of ‘Diagnose’ as its input. If both operations are mapped to value activities of ‘Eye-Care Specialist’ then ‘Result’ can be modeled by a value exchange between two value activities (see ⑥ in Fig. 5).

Guideline 7: The operation ‘Primary Treatment’ performed by ‘Primary Care Physician’ uses the entity (type of object) ‘Fee’ and produces the entity ‘Treatment’. The two dependencies ‘Achieve[Treatment for better Eyesight]’ and ‘Fee’ between ‘Patient’ and ‘Primary Care Physician’ involve these two entities. The operation ‘Primary Treatment’ is mapped to a value activity (guideline 5) and two dependencies introduce a value interface (guideline 6) in ‘Primary Care Physician’. Thus, we can draw value exchanges between the value interface of the value activity ‘Primary Treatment’ and value interface of the ‘Primary Care Physician’ (see ⑦ in Figure 5).

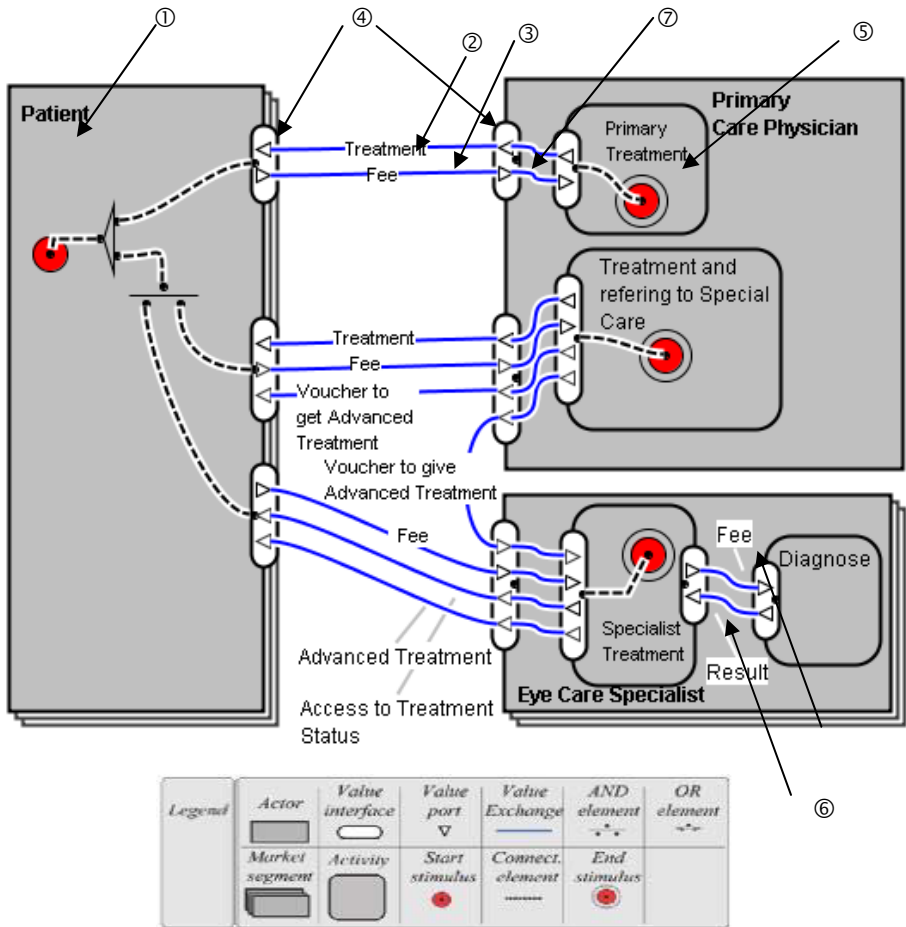


Fig. 5. e^3 value model for the health care case

7 Conclusion

In this paper we have addressed the problem of aligning business models with goal models. Business models, representing value exchanges among the involved actors, make it easier to justify design decisions at the IT level and trace them back to business levels. By enabling analysis of strategic business objectives, goal models may be used as a basis for designing alternative business models.

To address the outlined problem, we have proposed a requirement engineering method that starts by eliciting and analyzing strategic business goals using a goal model, and ends, using a set of transformations, with the creation of a goal-aligned business model. We have used well-established frameworks for presenting the two models, namely, KAOS for goal models and e^3 value for business models.

The proposed method offers a number of benefits. First, we have formulated the main elements of KAOS framework, such as goals and system requirements, in a uniform way and in terms of business concepts. Second, the proposed method provides a way of designing business models that are based on the goals and needs of an enterprise as expressed in a goal model. Third, using our method, it becomes possible to relate the components of a goal model to those of a business model, which gives a firm basis for establishing a complete traceability of decisions from the strategic to the operational level (where services and processes are designed).

A number of issues need to be addressed in future work. The main concern is the method validation, from the perspective of the designed business model. Currently, we have not established the conditions that the goal model needs to fulfill to enable the design of a self-standing business model.

References

1. Kotonya, G., Sommerville, I.: *Requirements Engineering: Processes and Techniques*. Wiley, Chichester (1998)
2. van Lamsweerde, A.: Goal-oriented requirements engineering: A guided tour, invited mini tutorial. In: *Proceedings of RE 2001 - International Joint Conference on Requirements Engineering*, Toronto (2001)
3. Gordijn, J., Akkermans, H.: Value based requirements engineering: Exploring innovative e-commerce idea. *Requirements Engineering Journal* 8(2), 114–134 (2003)
4. Gordijn, J., Akkermans, J.M., van Vliet, J.C.: What's in an electronic business model? In: Dieng, R., Corby, O. (eds.) *EKAUW 2000*. LNCS, vol. 1937, pp. 257–273. Springer, Heidelberg (2000)
5. McCarthy, W.E.: *The REA Accounting Model: A Generalized Framework for Accounting Systems in a Shared Data Environment*, *The Accounting Review* (1982)
6. Osterwalder, A.: *The Business Model Ontology. A Proposition in a Design Science Approach*, Ph.D-Thesis. University of Lausanne (2004)
7. UN/CEFACT Modelling Methodology (UMM), User Guide, http://www.unece.org/cefact/umm/UMM_userguide_220606.pdf (last accessed, October 2008)
8. Hraby, P.: *Model-Driven Design Using Business Patterns*. Springer, Heidelberg (2006)
9. Gordijn, J., Petit, M., Wieringa, R.: Understanding Business Strategies of Networked Value Constellations Using Value and Goal Modeling. In: *Proceedings of the 14th International Conference on Requirement Engineering, RE 2006* (2006)
10. Yu, E., Mylopoulos, J.: From E-R to A-R: Modelling Strategic Actor Relationships for Business Process Reengineering. *Int. Journal of Intelligent and Cooperative Information Systems* 4(2, 3) (1995)
11. Raadt, B., van der Jaap, G., Yu, E.: Exploring Web Services Ideas from a Business Value Perspective. In: *Proceedings of the 13th IEEE International Conference on Requirements Engineering (RE 2005)*, Los Alamitos, CA (2005)
12. Andersson, B., Bergholtz, M., Edirisuriya, A., Ilayperuma, T., Johannesson, P., Zdravkovic, J., Jayaweera, P.: Enterprise Sustainability through the Alignment of Goal Models and Business Models. In: *Proceedings of BUSITAL (a workshop on Business/IT Alignment and Interoperability)*, CAiSE 2008, Montpellier (June 2008)

13. Business Motivation Model release 1.3, The Business Rules Group (2007), http://www.businessrulesgroup.org/second_paper/BRG-BMM.pdf (last accessed, October 2007)
14. Andersson, B., Bergholtz, M., Edirisuriya, A., Ilayperuma, T., Johannesson, P., Zdravkovic, J.: Using Strategic Goal Analysis for Enhancing Value-based Business Models. In: Proceedings of the Workshop on Business/IT Alignment and Interoperability (BUSITAL 2007), Trondheim, Norway (2007)
15. Dardenne, A., van Lamsweerde, A., Fickas, S.: Goal-Directed Requirements Acquisition. *Science of Computer Programming* 20 (1993)
16. REMS (REMisslusS) Project, <http://www.rems.se> (last accessed, November 2007)
17. van Lamsweerde, A.: *Systematic Requirements Engineering - From System Goals to UML Models to Software Specifications*. Wiley, Chichester (2008)

Model-Based Penetration Test Framework for Web Applications Using TTCN-3

Pulei Xiong, Bernard Stepien, and Liam Peyton

School of Information Technology and Engineering,
University of Ottawa, Canada
{xiong,bernard,lpeyton}@site.uottawa.ca

Abstract. Penetration testing is a widely used method for testing the security of web applications, but it can be inefficient if it is not done systematically. Public databases of web application vulnerabilities can be used to drive penetration testing, but testers need to understand them and interpret them into executable test cases. This requires an in-depth knowledge of security. This paper proposes a model-based testing approach using a data model that describes the relationship between web security knowledge, business domain knowledge, and test case development. The approach consists of a data model that represents the relevance between attack surface, application fingerprint, attack vectors, and fuzz vectors; a test case generator that automatically generates penetration test scenarios for web applications; and a penetration test framework supported by TTCN-3 test environment. The model-based testing approach can be used to provide structured tool support for developing penetration test campaigns. We demonstrate the feasibility and efficiency of the approach at the design level.

Keywords: web application security, model-based testing, penetration testing, test specification, TTCN-3.

1 Introduction

Penetration testing is a widely used method for testing the security of web applications, but it can be inefficient if it is not done systematically. Public databases of web application vulnerabilities can be used to drive penetration testing, but testers need to understand them and interpret them into executable test cases. This requires an in-depth knowledge of security.

This paper proposes a model-based testing approach using a data model that describes the relationship between web security knowledge, business domain knowledge, and test case development. The approach consists of

- 1) a data model that represents the relevance between attack surface, application fingerprint, attack vectors, and fuzz vectors;
- 2) a test case generator that automatically generates penetration test scenarios for web applications;
- 3) and a penetration test framework supported by the TTCN-3 test environment.

The model-based testing approach can be used to provide structured tool support for developing penetration test campaigns. It is deemed to be more efficient, achieve higher test coverage, and conduct tests in a more consistent and systematic way. We demonstrate the feasibility and efficiency of the approach at the design level.

The remainder of the paper is organized as follows: Section 2 provides background knowledge on vulnerability assessment, penetration testing, and TTCN-3, and related work on web application penetration testing. Section 3 discusses a data model for penetration test case development. Section 4 discusses how to build a penetration test framework based on TTCN-3. Section 5 analyzes the advantages of the model-based approach. Section 6 concludes and indicates the direction of future work.

2 Background and Related Work

Vulnerability is a software or hardware bug or misconfiguration that a malicious individual can exploit [1]. Vulnerability assessment is the activities that identify security liabilities within a system (network, system software, and applications) [1], and verify that no known security vulnerability is present on the target system [2]. Penetration testing, also known as ethical hacking, is a process that goes one step further to substantiate the vulnerabilities reported during vulnerability assessment [1], by attempting to recreate the trickery and creativity that a real-live attacker would use [2]. While vulnerability assessment only reports vulnerabilities, penetration testing verifies that vulnerabilities actually exist. In the situations that confirmation of vulnerabilities is needed, penetration testing should be conducted.

Network vulnerability assessment follows the paradigm of “ports scanning, services enumeration, and vulnerabilities identification” [1]. Network vulnerability assessment identifies network vulnerabilities by directly referring to public vulnerability databases and mailing lists, e.g. the Open Source Vulnerability Database (OSVDB) [3], the CERT Vulnerability Notes Database [4], the Bugtraq mailing list [5], or by using vulnerability scanners such as Nessus [6] that themselves rely on the public vulnerability databases. Network vulnerability assessment is a structured approach that lends itself to automation due to its high degree of predictability and need for repeatability [2].

Vulnerability assessment and penetration testing are widely used in network security testing. However, security testing has moved beyond the realm of network [7] since applications, especially web applications, potentially introduce a whole new set of security vulnerabilities [2]. Therefore, with respect to a system running web applications, in addition to ensuring network and system software are secure, each web application itself has to be tested to make sure that the web application does not have any security vulnerabilities.

Academic research has been conducted on software security and penetration testing. In [7] and [8] the authors present risk-based software security and penetration testing methodology, and advocate to integrate the testing with the software development life cycle. In [9], the author presents a complete penetration process, starting with threat modeling. In [10], the author discusses the analysis of the threat and the potential attackers, which is valuable to conduct effective penetration testing.

Many literatures illustrate how to conduct penetration testing for web applications including [11], [12], and [13]. The common basic steps include:

- **Information gathering and discovery.** This is done by walking through a web application and observing its functionalities. In particular, all the HTTP requests (with their cookies and GET or POST variables) which represent the attack surface (entry points), are documented. Tools such as an intercepting proxy or a browser plug-in are needed to collect the requests. In this phase, the web application fingerprint can be identified. It includes the framework/language that is used to build the application, the backend database on which the application is based, and the operating system that the application runs on.
- **Test planning and design.** Attack scenarios are developed based on the entry points gathered in the previous phase. Vulnerabilities that potentially existing in each entry point are determined. Any cookie or variable that needs to be manipulated to conduct an attack is identified, and then is replaced by fuzz data.
- **Test execution.** Real attacks are carried out to verify potential vulnerabilities.

A few public vulnerability databases are recommended in [2], [9], [11], and [12] to help classify and identify web vulnerabilities and attacks, e.g., Common Vulnerabilities and Exposures (CVE) [14], Common Attack Pattern Enumeration and Classification (CAPEC) [15], Common Weakness Enumeration (CWE) [16], SANS Top-20 2007 Security Risks [17], OWASP Top-10 2007 web application vulnerabilities [18]. However, unlike those in network vulnerability assessment, the web application vulnerability databases can not be used in an automated way. The web application vulnerabilities are generic and might be present for any web application. For example, in many web applications, the “search” feature is potentially vulnerable to a reflected XSS attack; the “log in” feature is potentially vulnerable to a SQL injection attack. For a particular web application, these generic vulnerabilities should be substantiated by conducting penetration tests.

Apart from penetration testing, there are a variety of verification methods to be considered for web application security testing [11] [18], including automated methods such as a vulnerability scanner or static analyzer, and manual methods such as a code review. Table 1 shows the comparison of four verification methods applied to the top ten web application vulnerabilities [18]. We label a method as “Yes” if it is deemed to provide thorough coverage for a vulnerability, and “No” if it only provides partial coverage or no coverage at all. The result shows that the automated methods do not work as well for verification. They either do not work at all, or can only detect a limited subset of the cases for a vulnerability. The manual methods are more effective, but lack efficiency. The problem with code reviews is that, as a white-box method, it works at the source code level. In the cases where the code base is large, code reviews are time-consuming and error-prone. Penetration testing, as a black-box method, can be applied to most of the vulnerabilities, except the two vulnerabilities that are infrastructure-related. But when it is conducted in a manual way, it is time-consuming and might lack consistency [18].

Table 1. Verification Methods for OWASP Top Ten 2007

Vulnerability	Scanner	Analyzer	Code review	Pen-test
Cross Site Scripting (XSS)	No	No	Yes	Yes
Injection Flaws	No	No	Yes	Yes
Malicious File Execution	No	No	Yes	Yes
Insecure Direct Object Reference	No	No	Yes	Yes
Cross Site Request Forgery (CSRF)	No	No	Yes	Yes
Information Leakage and Improper Error Handling	No	No	Yes	Yes
Broken Authentication and Session Management	No	No	Yes (efficient when combined)	
Insecure Cryptographic Storage	No	No	Yes	No
Insecure Communications	No	No	Yes	No
Failure to Restrict URL Access	No	No	Yes (efficient when combined)	

In this paper, we will present, at the design level, how to build a model-based test framework using TTCN-3 that partially automates web application penetration testing to achieve efficient and consistent coverage in a systematic way. TTCN-3 is a standardized test specification and implementation language, developed by the European Telecommunications Standards Institute (ETSI) [19] for testing distributed systems. It provides powerful abstraction mechanisms for interfacing to different data and presentation formats and for defining test cases at different levels of abstraction, much as developers use modeling languages to specify the design of a system at different levels of abstraction. This enables reuse across different levels of test activities and the coordination and synchronization of test activities with development activities throughout the development life cycle.

3 Model-Based Penetration Testing for Web Applications

Penetration testing requires solid knowledge and skills on web security. As shown in Figure 1, a penetration tester develops test cases based on the knowledge of web security and the application itself (business domain). For each entry point (attack interface), he needs to determine in the context of application fingerprint:

- which vulnerabilities it potentially has, and accordingly which relevant attacks need to be applied
- for each attack scenario, which fuzz vector(s) can be applied

While the entry points and application fingerprints can be collected during the information gathering and discovery phase, the coverage of the test cases, and consequently the quality of the tests, are determined by (or limited to) the tester’s skills, knowledge, and experience on web security, as indicated in [8].

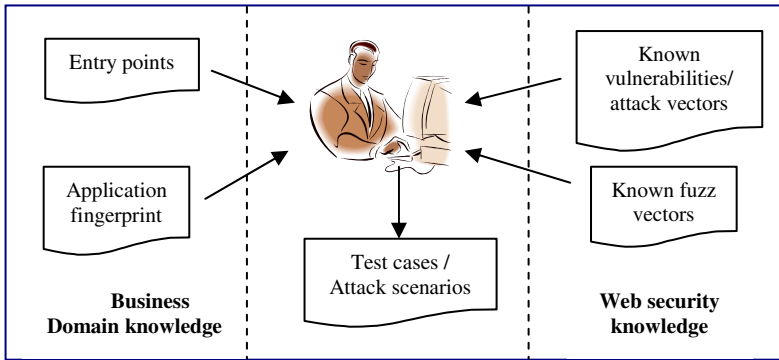


Fig. 1. Penetration Test case development

Although public source vulnerability databases are available to penetration testers to develop attack scenarios, they are text-based. The relevance of a vulnerability to a type of entry point is not specified in a well-structured way. They cannot be used to directly identify attack vectors for an entry point. Penetration testers have to understand all these vulnerabilities, determine relevant attacks and corresponding fuzz vectors, and then interpret them into executable test cases. This intelligent activity is a big obstacle to general testers. In addition, hackers probably have more resources than testers [10]. This means the battle between penetration testers and potential hackers is not fair – while a large number of hackers, who only need to be expert in a few security fields, have plenty of time to exploit any vulnerability in a web application, the testers have to detect all vulnerabilities, hopefully, before the web application is released and deployed into production, and always within an aggressive schedule. A cost-efficient approach becomes essential to web application penetration testing. If we can build a data model which represents the relevance between the business domain knowledge and the web security knowledge then a penetration tester can refer to the model to develop test cases. Furthermore, based on the data model, if we can automate the process of test case development and execution, even partially, it is definitely a step towards a more cost-efficient approach to penetration testing for web applications.

To understand the relevance between the business domain knowledge and the web security knowledge, and the relevant characteristics of web application penetration testing, we take a closer look at the web application vulnerabilities and attack vectors, as well as the fuzz vectors. We start our analysis with the Top Ten 2007 web application vulnerabilities [18], and the commonly used fuzz vectors documented in [11].

3.1 Analysis of the Top Ten Web Application Vulnerabilities 2007

The investigation of these vulnerabilities reveals the paradigm for conducting attacks, the relevance to web application frameworks, and the relevance to the functionality of an entry point:

- All penetration testing is performed via “URL manipulation” – manipulating HTTP requests including cookies, variables or URL paths.

- Although most vulnerabilities are applicable to all the web application frameworks, there are some exceptions. For example, PHP applications are particularly vulnerable to “Malicious File Execution”.
- Further analysis indicates that some vulnerabilities are relevant to specific application features. For example, “search” is very likely vulnerable to a reflected XSS attack; in a public forum, “viewing” contents that were entered by other users is vulnerable to a stored XSS attack; and password-based “Login” is vulnerable to a SQL injection attack.

3.2 Analysis of the Commonly Used Fuzz Vectors

Investigation on fuzz vectors reveals its classification and its relevance to the environment in which web applications run, such as the backend database, OS/file system, and encoding schema:

- The fuzz vectors are categorized by corresponding attack vectors, e.g. fuzz vectors for XSS, for Buffer Overflows (BFO), and for Format String Errors (FSE).
- Some fuzz vectors are database specific. For example, fuzz data of SQL Injection for Oracle, SQL Server, and MySQL might be different.
- Some fuzz vectors are OS specific. For example, on a Unix OS, root directory is “/”, and directory separator is “/”, while on a Windows OS, root directory is “<drive letter>:\”, and directory separator is “\” but also “/”.
- All fuzz vectors may be obfuscated using different encoding schema.

3.3 Data Model for Test Case Development

Based on the analysis on the Top Ten Web Application Vulnerabilities 2007 and the Commonly Used Fuzz Vectors, we built a data model shown in Figure 2. The data model describes the relationship between business domain knowledge, web security knowledge, and attack scenarios

- The HTTP requests including cookies, variables and URL path to be manipulated are essential to developing attack scenarios.
- The functionality related to an entry point can be used to identify relevant attack vectors.
- The identified attack vectors may be further filtered out based on the fingerprint of a web application, e.g. the platform (language) used to develop the web application.
- For each attack vector, corresponding fuzz vectors can be picked up based on the link between them.
- The corresponding fuzz vectors may be further filtered out based on fingerprint of a web application, e.g. the brand of the backend database, the operation system (and the file system) where the web application is running, and the encoding schema.

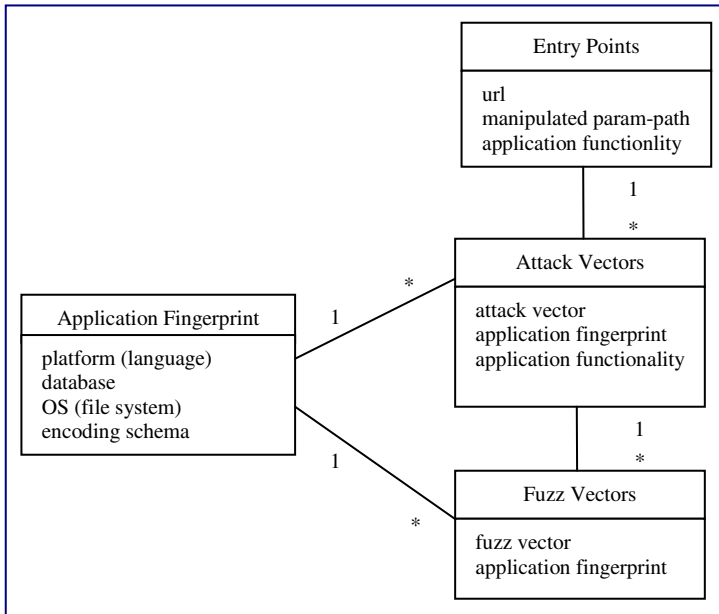


Fig. 2. Data model for Test case development

- The web application fingerprint consists of the characteristics that are relevant to filtering attack vectors and corresponding fuzz vectors, including platform (language), the brand of database, the type of OS (file system), and encoding schema.

3.4 Algorithm for Test Case Generator

Based on this data model, we can build a Test Case Generator (TCG) which can automatically generate test cases from entry points whose URL, manipulated parameter-path, and related application functionality have been specified, as shown in Figure 3.

The algorithm of the Test Case Generator is:

- For each entry point, identify relevant attack vectors from “Attack Vectors” table based on its functionality
- Filter out attack vectors from the identified ones based on web application fingerprint – platform (language)
- Pick up corresponding fuzz vectors from “Fuzz Vectors” table for each attack vector based on the link between attack vector and fuzz vector
- Filter out fuzz vectors from the selected ones based on web application fingerprint – database, OS (file system) and encoding schema
- Instantiate each attack vector based on entry point URL, and replace manipulated parameter-path with fuzz vector

In the next section, we will show how to build a penetration test framework using TTCN-3 to support the model-base penetration testing approach, and illustrate it using a reflected XSS test scenario as an example.

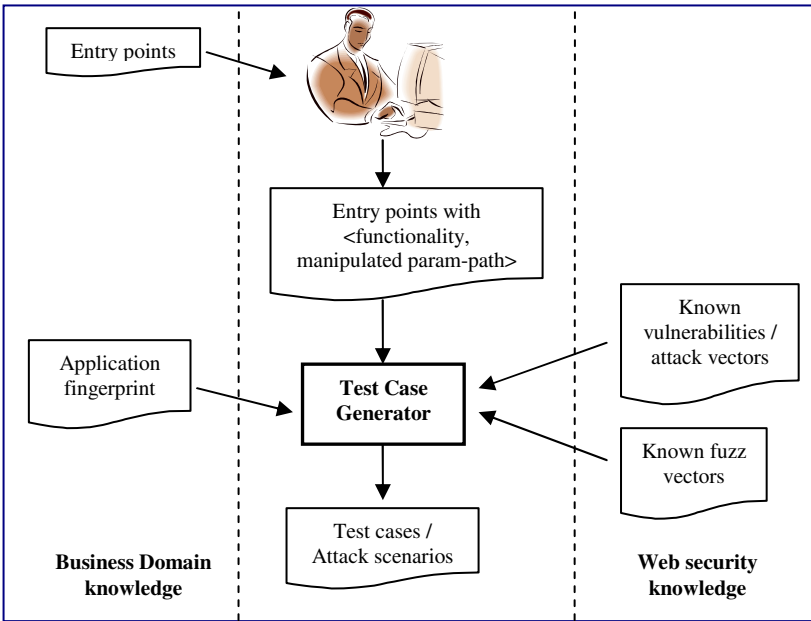


Fig. 3. Model-based Penetration Test case development process

4 Penetration Test Framework Based on TTCN-3

Basically, testing web applications is based on a request/response paradigm. That is, an HTTP request is sent to the application under test, and then the HTTP response from the application is checked. Testing web application security falls into the same paradigm. The test cases generated in Figure 3 are a set of HTTP requests. To run the test cases automatically, we need to specify the expected results for a request in the form of an HTTP response. We can achieve this by extending the “Fuzz Vectors” in Figure 2 with an expected HTTP response for each fuzz vector. The expected response is a 2-tuple as <expected value, html section of the expected value>. Accordingly, the generated test case is a 2-tuple as <http request, expected response>.

There was research done on web application and web service testing using TTCN-3 in [20], [21] and [22]. The research results show that TTCN-3 is a feasible platform for web application testing, with advantages such as test abstraction, powerful match mechanism, reusable adapters, and platform and language independent nature. We can build the penetration test framework based on TTCN-3, as showed in Figure 4. The test framework includes a set of pre-defined TTCN-3 templates for HTTP request (GET/POST), HTTP expected response, and HTML document, and a Test Case Engine which is built in TTCN-3, and runs on the standard TTCN-3 run time environment.

The algorithm of the engine is:

- Read each test case <request, expected response>
- Populate the pre-defined templates of HTTP request (GET/POST) and HTTP expected response using the test case data

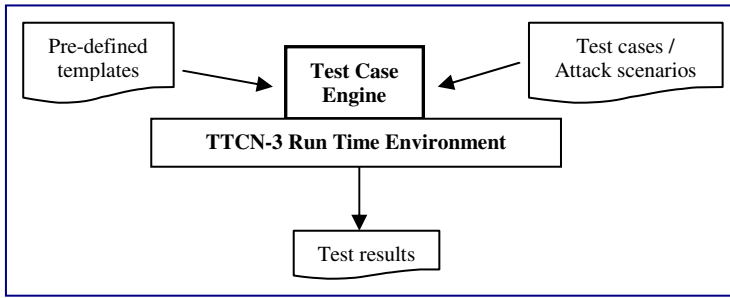


Fig. 4. Penetration Test Framework Based on TTCN-3

- Send out the HTTP request
- Wait for response and verify it against the expected response template
- Log test result if necessary

4.1 Test Scenario

We illustrate the whole process using a reflected XSS test scenario against a deliberately insecure J2EE web application, WebGoat, which is maintained by OWASP [23] as a standardized example. A XSS (Cross-site Scripting) attack is a type of HTML injection attack in which malicious scripts, embedded in the input to a web application, is injected into the web site, and then sent back to a different user (victim) in the form of browser-side scripts. A reflected XSS attack, also known as non-persistent XSS attack, is a type of XSS in which the injected malicious scripts is reflected (immediately) off the web server and displays (runs) on browser side. To make it simple, we do not consider “application fingerprint” in this example.

- As the first step, walk through WebGoat. Identify a scenario to conduct a XSS attack:
 - log in WebGoat
 - navigate to the page that is used to “search” a user
 - search for a known user and then observe the search result
 - log out
- Use an intercepting proxy, e.g. WebScarab [24], to gather all the relevant HTTP requests and responses for the scenario above
- Identify that the “search” request is an entry point, the HTTP method is POST, the field “username” is the field to be manipulated. Document the entry point with <functionality = “search”, param-path = “username”>
- Assume we already have the attack vector “reflected XSS” tagged with “function search” and linked with fuzz vector “<script>alert(“reflected XSS attack!”);</script>” with expected response “<script>alert(“reflected XSS attack!”);</script>”, “<body>” in place, we run the Test Case Generator using the entry point as input, we then get a test case as 2-tuple <manipulated request, response>
- Using the test case together with the complete scenario as input, we run the Test Engine, then we get a test result like: the web application WebGoat is

vulnerable to reflected XSS attack, since the malicious script “<script>alert("reflected XSS attack!");</script>” in search input is not detected and/or is not encoded appropriately in the search result returned by WebGoat

4.2 TTCN-3 Implementation Considerations

The basic criteria of a test oracle to detect an XSS vulnerability is to receive a status of 200 OK instead of an error code if the SUT would be XSS proof.

Since the input to the test case generator is the web application itself, the test generator has ample information to assemble an oracle beyond the 200 return code from the web response page. This information can then be specified to improve the oracle. With most general purpose programming languages, this could be a complicated task since the code needs to distinguish content from HTML formatting information. In TTCN3, however, the separation of concerns between an abstract view of the information and the concrete layout of the same information allows us to specify the oracle in a very concise way. In order to achieve this we however need two basic elements:

- an abstract data type to represent web information
- an HTML codec (coder decoder)

Both of these elements have already been addressed in our TTCN-3 web testing framework (described in [20], [21] and [22]). Thus, all we need to do is to generate TTCN-3 templates and write a simple re-usable request/response behavior part.

In our TTCN-3 web testing framework we have defined an abstract data type for web forms and web pages content as follows:

a web submit form for specifying our request:

```
type record FormElementType {
    charstring elementType,
    charstring name,
    charstring elementValue
}
type set of FormElementType FormElementSetType;
type record BrowseFormType {
    charstring name,
    charstring formAction,
    charstring kindMethod,
    FormElementSetType elements
}

```

```
type set of BrowseFormType FormSetType;
```

a web page to specify the response:

```
type record WebPageType {
    integer statusCode,
    charstring title,

```

```

    charstring content,
    LinkListType links optional,
    FormSetType forms optional,
    TableSetType tables optional
}

```

where some of the critical information could be the filled form itself containing the malicious XSS script.

In our WebGoat example, the generated request would be a series of templates to assemble BrowseFormType template as follows:

```

template FormElementType userName_t := {
    elementType := "TEXT",
    name := "Username",
    elementValue :=
"<script>alert(\"Dangerous\");</script>"
}
template FormElementType searchButton_t := {
    elementType := "SUBMIT",
    name := "SUBMIT",
    elementValue := "Search"
}
template BrowseFormType XSSAttackForm_t := {
    name := "form",
    formAction := "attack?Screen=13&menu=900",
    kindMethod := "POST",
    elements := { userName_t, searchButton_t }
}

```

Our response oracle doesn't need to specify all information in our template because some fields can be specified as either being any value which is represented in TTCN-3 by the "?" or "*" wildcards or using set operators such as superset to specify for example that among all forms present in the page, at least the form of interest containing the malicious script should be present.

```

template WebPageType XSS_attack_success_response_t := {
    statusCode := 200,
    title := "Phishing with XSS",
    content := pattern "No results were found.",
    links := ?,
    forms := superset (XSSAttackForm_t),
    tables := ?
}

```

Finally the test case itself could be parametrized. In a sense that would be a framework specific to XSS testing that could be re-used for a number of different tests.

```

testcase XSS_attack(template BrowseFormType
theRequest, template WebPageType theResponse) runs on
MTCType system SystemType {

```

```

map(mtc:webPort, system:system_webPort);

webPort.send(theRequest);
alt {
  [] webPort.receive(theResponse) {
    setverdict(fail)
  }
  [] webPort.receive {
    setverdict(pass)
  }
}
}

```

These tests can be fired up from the TTCN-3 control part as follows:

```

control {
    execute(XSS_attack(XSSAttackForm_t,
XSS_attack_success_response_t));
    ...
    execute(...)
}

```

5 Analysis of Results

Compared to the original test case development process in Figure 1, the process in Figure 3 is optimized by automatic test case generation. In this model-based test case development process, a tester has to explicitly specify requests (URL, parameter-path to be manipulated, and related application functionality for an entry point, but with the Test Case Generator test cases can be generated automatically. In the original process, the quality of penetration testing relies on the tester's expertise in web security, while in the optimized process, the quality of penetration testing relies on the correctness, timeliness, and completeness of the web security repository (known vulnerabilities/attack vectors, and corresponding fuzz vectors), which is deemed to provide consistent and systematic coverage. See Table 2 for the comparison of the two approaches.

Table 2. Original versus Model-based Process

Criteria	Original Process	Model-based Process
Efficiency	Manual process	Partial automated, deemed to be faster
Coverage	Limited to tester(s)' expertise on web security	Deemed to be higher
Quality	Limited to tester(s)' diligence	Consistent and systematic
Limitation	The availability of a highly-skilled test team	The availability of a high quality (correctness, timeliness, and completeness) web security repository

As for implementing the test framework using TTCN-3, we have not conducted a systematic comparison with other alternatives such as using a programming language like Java or a scripting language. Based on our previous research on web application testing using TTCN-3, we think TTCN-3 is a feasible test framework to support the model-based approach to web application penetration testing. As a future work, we will complete the TTCN-3 implementation, explore other alternatives, and compare them.

6 Conclusions and Future Work

In this paper, we propose a model-based approach to web application penetration testing using TTCN-3. We present the approach at the design level, supplemented with a sample XSS test scenario. We also compare the model-based approach with the manual test approach. We conclude that the model-based approach is deemed to be more efficient, with higher coverage, and produce better quality of tests.

In future work, we will

- Conduct a thorough investigation on web application security, and build a more complete web security knowledge repository.
- Complete the implementation based on TTCN-3, and conduct a case study using the model-based test framework against a typical web application.
- Consider building up attack scenario model for a web application using model language UCM (use case map), and explore the possibility of deriving security test scenarios specified in TTCN-3 from the UCM model.

Acknowledgements

The authors would like to thank Testing Technologies IST GmbH for providing us the necessary tool -- TTworkbench to carry out this research as well as NSERC for partially funding this work.

References

1. Manzuik, S., Gold, A., Gatford, C.: Network Security Assessment: From Vulnerability to Patch. Syngress Publishing (2007)
2. Splaine, S.: Testing Web Security: Assessing the Security of Web Sites and Applications. John Wiley & Sons, Chichester (2002)
3. Open Source Vulnerability Database (OSVDB), <http://osvdb.org/>
4. CERT Vulnerability Notes Database, <http://www.kb.cert.org/vuls/>
5. Bugtraq mailing list, <http://www.securityfocus.com/archive/1>
6. Nessus vulnerability scanner, <http://www.nessus.org/nessus/>
7. Potter, B., McGraw, G.: Software Security Testing. *IEEE Security & Privacy* 2(5), 81–85 (2004)
8. Arkin, B., Stender, S., McGraw, G.: Software Penetration Testing. *IEEE Security & Privacy* 3(1), 84–87 (2005)

9. Thompson, H.: Application Penetration Testing. *IEEE Security & Privacy* 3(1), 66–69 (2005)
10. Bishop, M.: About Penetration Testing. *IEEE Security & Privacy* 5(6), 84–87 (2007)
11. OWASP TESTING GUIDE Version 3.0, OWASP Foundation (2008)
12. Andreu, A.: Professional Pen Testing for Web Applications. Wrox Press (2006)
13. Palmer, S.: Web Application Vulnerabilities: Detect, Exploit, Prevent. Syngress Publishing (2007)
14. Common Vulnerabilities and Exposures (CVE), <http://cve.mitre.org>
15. Common Attack Pattern Enumeration and Classification (CAPEC), <http://capec.mitre.org>
16. Common Weakness Enumeration (CWE), <http://cwe.mitre.org>
17. SANS Top-20, Security Risks (2007), <http://www.sans.org/top20/>
18. OWASP TOP Ten (2007), http://www.owasp.org/index.php/Top_10_2007
19. ETSI ES 201 873-1, The Testing and Test Control Notation version 3, Part1: TTCN-3 Core notation, V3.4.1 (September 2008)
20. Probert, R.L., Xiong, P., Stepien, B.: Life-cycle E-Commerce Testing with OO-TTCN-3. In: FORTE 2004 Workshops proceedings (September 2004)
21. Stepien, B., Peyton, L., Xiong, P.: Framework Testing of Web Applications using TTCN-3. *International Journal on Software Tools for Technology Transfer* 10(4), 371–381 (2008)
22. Xiong, P., Probert, R.L., Stepien, B.: An Efficient Formal Testing Approach for Web Service with TTCN-3. In: Proc. of the 13th International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2005) (September 2005)
23. OWASP WebGoat Project, http://www.owasp.org/index.php/Category:OWASP_WebGoat_Project
24. OWASP WebScarab Project, http://www.owasp.org/index.php/Category:OWASP_WebScarab_Project

Impact of Diversity on Open Source Software

Hiba Enayat, Steven Muegge, and Stoyan Tanev

Department of Systems and Computer Engineering, Carleton University
1125 Colonel By Drive, Ottawa ON K1S 5B6 Canada

Abstract. This paper examines the relationship between open source project diversity and success. The sample of open source projects includes all mature projects driven by the Eclipse Foundation as of February 2008. Three types of project diversity were used: i) organizational – measured by number of committers per organization per project, ii) contribution – measured by the number of commits made per organization per project, and iii) technical – measured by the number of commits made per given software file type. Success was measured by means of i) economic metrics, including the number of corporate adoptions and the number of jobs postings including the project name, and ii) development metrics, including the project popularity and the growth of the intensity of members' activity. The paper makes two main contributions. First, we contribute to the literature on open source software and diversity. Second, we introduce economic success metrics to the empirical assessment of open source software project success.

1 Introduction

The *Oxford English Dictionary* (1989) defines "diversity" as "the condition or quality of being diverse, different, or varied; difference, unlikeness." It implies groups with differences between group members. Aspects of diversity often examined in management research include differences in age, education, firm tenure and functional or technical background (Tatli & Ozbilgin, 2005; Knight et al. 1999), all of which are empirically associated with enhanced group creativity and innovativeness, as well as company success (Winston, 2001). Prior research shows that many of these factors are within the control of a firm (Auh & Menguc, 2005)

Community-based open source software (OSS) projects, such as the Linux Kernel, are the result of diverse contributors collaborating to produce high performing products. Hence, diversity may be an important variable to consider for practitioners seeking increased creativity and innovation in OSS projects.

2 Diversity

Systems researcher Scott Page (Page, 2007) provides a unified rationale and framework for studying diversity. According to Page, homogenous groups tend to communicate in common language and have similar training and experiences, so they tend to approach problems in the same manner. Also, their choice of common language alienates others who are not familiar with the same terms (q.v. Dougherty, 1992). Stating problems differently allows different problem-solving approaches to be used.

Page employs a *toolbox metaphor* to conceptualize the benefits of diversity. Tools enable people to solve problems and the right tools help get a job done faster. People have toolboxes with a diverse set of tools developed from experiences and training. The tools of problem solving are *heuristics*. Diverse teams with a greater variety of tools can mix and match tools to provide new combinations to find new ways to solve problems.

One potential problem with diversity is *preference difference*. Page distinguishes between instrumental and fundamental preferences. Instrumental preferences denote the same goals but a difference in the method used to establish the goal. Fundamental preferences denote differences in the end goal, and it is differences in fundamental preferences that can cause severe friction in groups.

3 Diversity and Organizations

Table 1 and Table 2 summarize empirical findings from the literature on diversity. Both positive and negative effects of diversity are reported, and there are many examples of conflicting effects on performance. In summary, diversity leads to enhanced creativity, increased performance and innovation. However, it also leads to poor implementation of the 4 C's (communication, collaboration, coordination and cohesiveness) (Auh & Menguc, 2005). Also, when people have diverse preferences, each problem has different solutions for each problem solver, which can result in preference cycles, misrepresentation, and manipulation (Page, 2007). Some relationships between diversity and associated benefits depend on the presence of moderating factors. For instance, top management team education and experience diversity led to innovativeness when inter-functional coordination was high (Auh & Menguc, 2005). This denotes that diversity provides beneficial results, but under certain circumstances.

Table 1. Empirical findings on positive effects of diversity

Independent Variable	Dependent Variable	Source
TMT employment tenure	Strategic consensus	Knight et al. (1999)
TMT interpretive ambiguity	Firm performance	Kilduff et al. (2000)
Diversity of academic organizations	Success	Winston (2001)
Functional diversity of teams	i) Innovation ii) Clarity of strategies iii) Ability to compete	Bunderson & Sutcliffe (2002)
Intrapersonal diversity of teams	i) Information sharing ii) Unit performance	Roberson & Park (2004)
Workforce diversity	i) Customer relations ii) Market share iii) Employee relations iv) Quality of workforce v) Reduced labour cost	Tatli & Ozbilgin (2005)
Organizational tolerance (diversity management)	Merger performance	Bellinger & Hillman (2000)
i) Organizational tenure ii) Educational iii) TMT iv) TMT education and experience	i) Inter-functional interaction ii) ROI and sales growth iii) Administrative innovation iv) Innovativeness	Auh & Menguc (2005)
Moderate tech. diversity in R&D alliances	Firm innovation	Sampson (2007)

Table 2. Empirical findings on negative effects of diversity

Independent Variable	Dependent Variable	Source
i) Age ii) Functional	i) Team agreement ii) Interpersonal conflict	Knight et al. (1999)
Workgroup diversity	Psychological attachment to org.	Kilduff et al. (2000)
Functional	i) Team effectiveness ii) Competitive response iii) Performance iv) Increased conflict v) Information sharing	Bunderson & Sutcliffe (2002)
Workforce diversity with lack of diversity management	i) Morale ii) Ambiguity iii) Increased conflict iv) Confusion v) Communication	Tatli & Ozbilgin (2005)
i) Organizational tenure ii) Functional iii) TMT Functional	i) Group performance after certain level & conformance to individual average ii) Consensus; iii) Innovativeness	Auh & Menguc (2005)

Although most studies employed linear models, some researchers have identified non-linear relationships between diversity and other variables (Auh & Menguc, 2005). Organizational tenure increased inter-functional interaction, but it led to deterioration of group performance beyond a certain length. R&D alliances with moderate technical diversity contributed more to firm innovation than lower and higher levels of diversity. Non-linear associations, such as the “inverted U- shaped curve” have been found in other areas of inquiry, such as structure and performance (Davis et al. 2007), the importance of customer input and product newness (Callahan & Lasry, 2004), and information processing and task complexity (Schroder et al. 1967).

Looking across the literature surveyed, we note the following trends about diversity in organizations:

- In comparison to homogeneity, diversity is associated with increased creativity and improved problem solving abilities.
- Too much diversity causes problems, such as lack of team cohesion, poor implementation of 4 C’s and others that can be considered an overall drop in performance.

These trends appear similar to those observed for the impact of structure on performance or of information processing on task complexity (Callahan & Lasry, 2004; Schroder et al. 1967). Since diversity primarily benefits by increased creativity, it would likely play an important role in performance improvements and innovation. However, with increased diversity, more preference differences are likely, causing miscommunication, slow response and misrepresentation (Page, 2007), poor 4Cs implementation and hence an overall decrease in performance. This hints at a “inverted U” relationship between diversity and performance, with an optimal level of diversity providing maximum performance.

4 Diversity and OSS Projects

OSS projects are the result of the efforts of diverse participant groups. Roles include passive users, readers, bug reporters, bug fixers, peripheral developers, active developers, core developers, project leaders, list participants, gate keepers and paid developers (Gallivan, 2001; Farraro & O'Mahony, 2003; Nakakoji et al. 2002; Stewart & Ammeter, 2002; von Krogh, 2003). OSS projects build upon collective efforts and can thus have numerous modules (Tuomi, 2001) in various platforms. Furthermore, OSS project communities are virtual development communities, distributed geographically (Hinds & Lee, 2008). Also, with the emergence of sponsored OSS projects (West & O'Mahony, 2005), organizations can be involved in the development and growth of the project. The contribution ratio in open source project is typically quite skewed; Bonaccorsi et al (2003) found that 10% of the developers in a project produced 70% of the code. Given this, we can at least identify the following kinds of diversity applicable to OSS projects: geographic and demographic, functional, educational, age, experience, contribution ratio, organizational (sponsored OSS projects) and technical content (modules and type of technology used). Hence, OSS communities are diverse in many ways, with each aspect of diversity potentially impacting OSS success.

5 Hypothesis Development

5.1 Research Question and Research Model

The main research question of this study concerns the relationship between OSS project diversity and success. Fig. 1 shows the hypothesized relationships between variables. To make meaningful propositions regarding the role of diversity in OSS success, we limit the scope and type of diversity to functional diversity (Bunderson & Sutcliffe, 2002), with project duration as a moderating factor. From the literature reviewed in sections III and IV, we expect the breadth of functional experience across team members to bring in diverse perspectives and interpretations, resulting in enhanced problem solving and creativity. However, functionally heterogeneous teams will likely face communication problems as specialists employ terms familiar within their functional domain that may be unfamiliar to others.

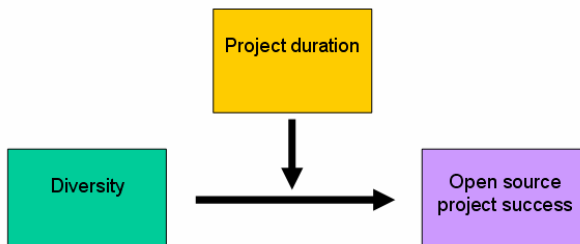


Fig. 1. Hypothesized relationships between OSS project diversity and project success

Other barriers may include lack of information sharing as one specialist may be reluctant to share information with others as they may not fully understand it and may not appreciate the impact it may have on other functional areas. In a functionally homogeneous team, members are from the same functional area and will likely only consider aspects most familiar to them and approach problems in the same way, thus leading to less than optimal solutions. As team functional diversity grows, it would bring in enhanced creativity and problem solving capacity, helping in ways such as considering more product features and different implementation possibilities, thus producing a more innovative solution. However, as team functional diversity grows, the distance to cover to communicate effectively to all members would grow. Information sharing would decline as members become less able to comprehend other members.

Applying the toolbox framework, team members would possess tools that are so diverse that mixing and matching tools takes longer and may not always be easily combined. This could result in increased response time, as each member takes longer to understand others and could eventually result in reduced innovation.

Crowston et al. (2006) propose a variety of open source project success metrics, including popularity, jobs acquired through involvement in project and level of activity. Our own previous work shows the particular usefulness of economic success metrics (Enayat, 2008). One economic success metric found in Crowston et al (2006) is the percentage of job postings requiring skills associated with a given OSS project or product.

5.2 Hypotheses

Popularity is generally an indication of focus of user attention based on number of subscribers and web hits (Stewart & Ammeter, 2002). The number of search engine hits on the project name is possible measure of popularity (Weiss, 2005).

We expect popularity to increase with functional diversity, as the project receives more attention and reaps benefits of increased creativity, resulting in incremental improvements to the project, and thus attaining more user focus and receiving more web publications. These publications may not be removed, but instead remain available as archives; thus we expect the effect of increasing diversity to saturate as the adverse effects of diversity start to appear. We therefore propose the following hypothesis:

Hypothesis 1: The level of popularity will increase, but reach saturation with the level of diversity.

Level of activity is an indication of the development effort put in by the members of a project. A simple measure of the level of activity could be the growth of the number of developers contributing to a given project, measured by mining code contribution, bug reports and mailing lists (Crowston et al. 2006).

We expect projects with low functional diversity to have fewer developers, like core developers, laid out in a style and architecture they are familiar with. Under such circumstances, we expect the level of activity of these developers to be high, with few incentives to entice new external contributors. As the project diversity grows, it would be architected in such a way to promote collaborative efforts from external contributors, thus reaching even more increased levels of activity. However, beyond a certain

point, we expect the adverse effects of diversity to manifest, resulting in lower activity levels, since the coordination and communication efforts required to sustain growth rate would be more taxing, with effects similar to figure 1. Thus we propose the following hypothesis:

Hypothesis 2: The level of activity will increase, reach an optimum and then decline with levels of diversity.

Job postings including project names indicate an interest by commercial parties looking to hire developers experienced with a given OSS project. These can be measured by quantifying job trends by project name on Indeed.com (Enayat, 2008). When OSS is high performing, it will likely be well adopted, thus creating a need for companies to hire people experienced with the software. If, however, there is a drop in the OSS performance, the adoption is likely to decrease and perhaps even existing adopters may consider new avenues. There is however, a time dependency, similar in nature to that of popularity, so if considering the number of job postings for a given project, it is expected to have a similar “inverted U” shaped behavior, but likely a more skewed shape, indicating slow growth after a certain point or saturation. Thus we propose the following hypothesis:

Hypothesis 3: The level of job postings with project names will increase, but reach saturation with the level of diversity.

The relationship between project diversity and OSS project success seems to be a dynamic phenomenon changing from incubation, early and mature stages of the project. Since diversity helps most in times when increased creativity is most needed, it is expected that its relevance will be more evident at the early project stages, i.e. for younger projects. Hence we propose the following hypothesis:

Hypothesis 4: The relationship between success and diversity will be more statistically significant for younger projects.

6 Research Method

6.1 Unit of Analysis: Project Driven by the Eclipse Foundation

The Eclipse Foundation governs the Eclipse open source community. The Eclipse platform comprises extensible frameworks, tools and runtimes for building, deploying and managing software across the development lifecycle. The Eclipse membership is a large and vibrant ecosystem of major technology vendors, innovative start-ups, universities, research institutions and individuals.

6.2 Study Period

The time period for this study was from the beginning of the project entry in the Eclipse commits log database until February 28th, 2008, for all mature Eclipse projects.

6.3 Sample

The study sample is a census of all mature Eclipse projects as of February 2008. Within Eclipse, there are two types of project: top-level projects and sub-projects. Sub-projects fall under a top-level project and are often a module of the top-level project. At the end of the study period, there were 11 top-level projects posted on the Eclipse.org website, of which 9 were mature. Most of these top-level projects have sub-projects. Table 3 lists all the projects used in this study and identifies their formal project category.

Table 3. Sample projects and formal category level

Projects	Project Level
Business Intelligence and Reporting Tools (BIRT)	Top-level
Data Tools Platform	Top-level
ERCP - Embedded Rich Client Platform	Sub-project
Target Management	Sub-project
Eclipse Project	Top-level
Eclipse Modeling Project	Top-level
Eclipse Communication Framework Project	Sub-project
Eclipse Process Framework Project	Sub-project
Phoenix Project	Sub-project
Rich Ajax Platform	Sub-project
Voice Tools Project	Sub-project
AJDT - AspectJ Development Tools Project	Sub-project
AspectJ	Sub-project
Buckminster Component Assembly	Sub-project
C/C++ Development Tooling (CDT)	Sub-project
COBOL	Sub-project
GEF - Graphical Editor Framework	Sub-project
Mylyn Project	Sub-project
Eclipse Orbit Project	Sub-project
PHP Development Tools	Sub-project
Parallel Tools Platform (PTP)	Sub-project
VE - Visual Editor	Sub-project
Test and Performance Tools Platform Project	Top-level
Eclipse Web Tools Platform Project	Top-level

6.4 Research Variables

In all cases, diversity was quantified using Blau's (1977) index of diversity, with a score of 0 corresponding to a homogenous group and 1 corresponding to a heterogeneous group. For the purpose of this research we have considered three types of diversity: organizational, contribution and technical content diversity. The measurement of the corresponding variables was found in the rich database tool called DASH – a commits log database that records submission-based details for Eclipse projects. The data stored include change, login name of committer, top and sub project information, file type of change and the company the committer belongs to. The limitations associated with this database are as follows:

- Some of the user names were changed when OTI (Object Technology International Inc.) merged into IBM. Commits by the old login should be classified under the new. These old logins are categorized as “unknown.”
- People who have changed companies (from A to B) are being reported as having worked for their current company (B) for all of the record history rather than as having worked for company A first, and then for company B.
- People categorized under “individual” may be associated with a company, just not an Eclipse member company that has signed a Member Committer Agreement. These data may require cleaning up.

Since the origin of the “individual” and “unknown” types requires extensive investigation, these categories will be used "as is" for the purposes of this study. Given the low frequency of occurrence of the above types, the impact on the results is expected to be small.

6.4.1 Organizational Diversity

Organizational diversity (OD) was measured by the number of committers per organization per project. To put this in perspective, consider the example of a project which has 4 companies, A, B, C and D with 2, 3, 5 and 10 committers respectively. The OD value for this project would then be as follows:

$$OD = 1 - \sum ((2/20)^2 + (3/20)^2 + (5/20)^2 + (10/20)^2) = 0.6550$$

To measure OD for a given project, a two steps were taken: i) queries were run to identify the number of organizations associated with a given project, ii) queries were run to identify the number of contributors of a given organization for that specific project.

It is important to note that external members, who may not be part of one of the member organizations, would fall into the ‘individual’ or ‘unknown’ category in the commits log database under the company field. The calculation for OD has also incorporated these fields and hence accounts for external contributors.

6.4.2 Contribution Diversity

Contribution diversity (CD) was measured by the number of commits made by organizations per project. To put this in perspective, consider the example of a project

which has four participant companies, A, B, C, and D that have made 200, 300, 500 and 1000 total commits, respectively. The CD value for this project would then be as follows:

$$CD = 1 - \Sigma ((200/2000)^2 + (300/2000)^2 + (500/2000)^2 + (1000/2000)^2) = 0.6550$$

To calculate the contributions each member firm made to a given project, a query was run to identify the number of commits made per company. It should be noted that there are limitations to quantifying CD in this way since a commit can be of variable lengths - one line of code and hundred lines of code submitted by two different developers would still count as equivalent types of commits. However, on the whole, it is a good indicator of the contribution activity and distribution between the groups related to a project.

6.4.3 Technical Content Diversity

Technical content diversity (TD) was measured by the number of commits made per given file type. To put this in perspective, consider the example of a project which has submission of types C++, Java, *.php and *.xml in the order of 20, 30, 50 and 100 commits, respectively. The technical content diversity value for this project would then be as follows:

$$TD = 1 - \Sigma ((20/200)^2 + (30/200)^2 + (50/200)^2 + (100/200)^2) = 0.6550$$

To calculate this value, queries were run to identify the number of commits made grouped by file type. This displayed all the resulting file types used and the number of commits made with each file type. We found that some of the developers have put in ambiguous entries for file types, such as numbers from 1 through 4. These did not provide any useful information on the technical characteristics of the commit and hence were excluded from the analysis.

6.4.4 Job Postings

Eclipse managers indicated a significant interest in economic success metrics and identified Indeed.com as a tool that could be used to identify the number of job postings, over time, containing the project name as phrase word. Interestingly, this measure has also been internally used by Eclipse as a performance metric. To gather this information, searches were done on indeed.com's job trends section, with the project name as the search criteria. The resulting graph was saved and the area under the curve was used as a quantifying method to provide a single 'Jobs' value for each of the projects.

6.4.5 Level of Activity

The level of activity associated with a given OSS project was identified as another success metric and was measured by the growth of the number of developers associated with that project. Calculating growth in developers' activity was a multi step process. Queries were run for each project to identify the unique logins per month from the start of the project until February 2008. This was organized by month. Growth charts were generated that mapped number of logins over time. It was observed that in some cases, the curve was not linear. Under these circumstances, it

was found appropriate to use linear regression to model the growth of the number of developers over time, i.e. approximate the growth curve linearly, and use the gradient of the line as the final growth metric.

6.4.6 Popularity

OSS project popularity was identified as the third success metric. We have followed Weiss (2005) in measuring popularity by using web search engines to create three popularity metrics - phrase queries (PQ), link queries (LQ) and site queries (SQ). A phrase query represents the number of hits with exact project name, a link query represents the number of external sites linked to a given OSS project page, and a site query represents the number of internal pages linked to a given project web page. The searches were conducted on Google web search engine. A phrase query returned the number of hits with exact project name matches. A link query returned the number of pages linked to this projects web page and a site query returned the number of web pages within the same domain, that were linked to this page.

6.4.7 Data Analysis and Hypothesis Testing

SPSS allows using multiple independent variables (OD, CD and TD) to test for relationships with one dependent variable (popularity, activity or job postings including project name) by means of stepwise linear regression (SLR) resulting in a model equation with the best predictive power. SLR was used to test all the hypotheses developed in this study. To perform a SLR, all the variables have to be normally distributed. To decide whether or not a variable was normally distributed, its kurtosis and skewness statistics were examined. A variable was considered to be normally distributed when its skewness and kurtosis statistics were less than twice their standard errors. Based on this reasoning the 'Jobs' variable was transformed by means of square root operation to $\sqrt{\text{Jobs}}$. In addition, all popularity variables PQ, LQ, and SQ, were transformed by means of a decimal logarithm operation to $\log_{10}(\text{PQ})$, $\log_{10}(\text{LQ})$ and $\log_{10}(\text{SQ})$.

To test hypothesis 1, each of the three normalized popularity dependent variables were regressed individually with all the independent variables. In addition, the total diversity variable ($\text{TOT} = \text{OD} + \text{CD} + \text{TD}$) was regressed with all the independent variables. Hypothesis 2 was tested using normalized activity growth dependent variable with all the independent variables. Hypothesis 3 was tested by regressing normalized jobs dependent variable with all the constructed independent variables. To test hypothesis 4, all the above tests were conducted for two different situations: i) for all the projects together, ii) for projects with duration less than 38 months (the median value of all project durations), and iii) for projects with duration larger than 38 months.

To conduct the analysis, SPSS was used to run stepwise linear regressions of the diversity metrics as independent variables and success metrics as the independent variables. To capture any possible non-linear effects of diversity, squares and cubes of the diversity variables were also employed as independent variables. Only statistically significant results with $p < 0.1$ were considered.

7 Research Results

Table 4 below includes all the statistically significant relationships that were found between the dependent and the independent variables together with their p-values and

explanatory power (R^2 in the case of single independent variables and Adjusted R^2 in the case of multiple independent variables). The last column also shows the hypotheses that were (+) or were not (-) supported by these relationships. Surprisingly, the only statistically significant relationships found were the ones involving the 'Job' success metric, i.e. the economic metric of project success.

Table 4. Summary of all statistically significant relationships between OSS project diversity and success

No	Case	Diversity variables (independent)	Success variables (dependent)	Adj. R^2	R^2	$p <$	Hypothesis
1	All projects	+ OD & + TD	sqrt(Jobs)	0.325		0.01	(+/-) H3
2	All projects	+ TOT ² & - CD ³	sqrt(Jobs)	0.422		0.005	(+) H3
3	Projects with T<=38	+ OD ²	sqrt(Jobs)		0.330	0.05	(+) H4

The first relationship shown in Table 4 is based on a multiple linear regression model with 32.5% explanatory power corresponding to a positive association between the normalized number of job postings sqrt(Jobs) and both, organizational (OD) and technical content (TD), diversity. It supports Hypothesis 3 as far as it shows that the normalized number of job postings is positively associated with organizational and technical content diversity but fails to support the expected saturation effect.

The second relationship in Table 4 is based on a multiple nonlinear regression model with 42.2% explanatory power and can be equated as follows: $\text{sqrt}(\text{Jobs}) = -0.178 + 0.194 * (\text{TOT})^2 - 1.117 * (\text{CD})^3$. The range of (TOT = OD+CD+TD) is between 0.66 and 2.18. The range of CD is between 0 and 0.8. A glance at the equation shows that it describes the increase of the dependent variable sqrt(Jobs) based on the increment of the quadratic term TOT² but then, at a certain point, demonstrates a saturation of that increase. The saturation is based on the negative effect of cubic term CD³ which has a smaller range of variation. The nonlinear model is a better approximation (i.e., with a higher explanatory power) of the actual relationship between the normalized job posting economic success variable and the total diversity of the OSS projects. The nonlinear model fully supports Hypothesis 3.

The third relationship shown in Table 4 is again based on a nonlinear regression model showing that organizational diversity has an impact on the normalized number of job postings for younger projects. This relationship is in full support of Hypothesis 4.

There were no statistical significant results in support of Hypotheses 1 and 2.

8 Summary of Key Findings

We report the following four insights from this analysis:

- 1) Economic success metrics were the only ones that were found to be statistically significant for the OSS projects in the research sample.

The only statistically significant relationship found for all cases had to do with the economic metric associated with the number of job postings containing the OSS project name. Interestingly, this success metric was found to be one of the most relevant to the Eclipse Foundation.

- 2) Organizational diversity is more relevant for younger projects (less than 38 months).

Organizational diversity is important for the economic success (normalized number of job postings) of younger OSS projects, i.e. projects that are at their relatively early maturity stages. This finding is in agreement with published literature.

- 3) Project popularity was found to be statistically insignificant for all projects.

Project popularity, which was determined on basis of web search hits on Google, was not found to be a statistically significant success metric and did not contribute to the insights generated by this study. The methodological aspects of the way it was measured need to be further studied.

- 4) There is a highly statistically significant, positive nonlinear relationship, with a relatively large explanatory power, between the number of job postings including the project name the total diversity of OSS projects.

This final insight is the most important result of our research study.

9 Conclusions

In this paper we have presented the results of a research study examining the relationship between OSS project diversity and success. Its main contribution lays in the demonstration of the statistical significance of economic measures, such as the number of job postings related to specific Eclipse software products, for measuring project success. This finding confirms the expectations of the Eclipse Foundation that economic measures should and must play a significant role in evaluating project success. The results have indicated the existence of a non-linear relationship between project diversity and project success showing that there is much to be learned about managing project diversity. It has also indicated that organizational diversity is important for the economic success of younger projects and hence incubators should try to encourage more organizational involvement early on in the project life cycle. In addition to introducing an economic success metric in quantifying OSS project success, we believe to have also validated a definition of the “Job” success metric that could be of methodological interest in future research studies.

Finally, it should be noted that this study is based on sponsored OSS projects, which tend to be more successful than non-sponsored OSS projects, thus limiting the full applicability of results and insights to non-sponsored OSS projects. For non-sponsored OSS projects, development type success metrics may be more appropriate to understand the relationship between diversity and success.

References

- Auh, A., Menguc, B.: Top management team diversity and innovativeness: the moderating role of inter-functional coordination. *Industrial Marketing Management* 34, 249–261 (2005)
- Bellinger, L., Hillman, A.J.: Does tolerance lead to better partnering? The relationship between diversity management and M and A success. *Business and Society* 39(3), 323–337 (2000)

- Blau, P.M.: *Inequality and heterogeneity*. Free Press, New York (1977)
- Bonaccorsi, A., Rossi, C.: Why open source software can succeed. *Research Policy* 32(7), 1243–1258 (2003)
- Bunderson, J., Sutcliffe, K.: Comparing alternative conceptualizations of functional diversity in management teams: process and performance effects. *Academy of Management Journal* 45(5), 875–893 (2002)
- Callahan, J., Lasry, E.: The importance of customer input in the development of very new products. *R&D Management* 34(2), 107–120 (2004)
- Crowston, K., Howison, J., Annabi, H.: Information systems success in free and open source software development: theory and measures. *Software Process: Improvement and Practice* 11(2), 123–148 (2006)
- Davis, J.P., Eisenhardt, K.M., Bingham, C.B.: Complexity theory, market dynamism and the strategy of simple rules. *Stanford Technology Ventures Program working paper* (2007) <http://web.mit.edu/~jasond/www/complexity.htm> (accessed January 6, 2009)
- Dougherty, D.: Interpretive barriers to successful product innovation in large firms. *Organization Science* 3(2), 179–202 (1992)
- Enayat, H.: Relationship between diversity and open source software success. *Technology Innovation Management Program Project*, Carleton University, Ontario (2008)
- Oxford English Dictionary, 2nd edn. Oxford University Press, Oxford (1989)
- Farraro, F., O'Mahony, S.: Managing the boundary of an 'open' project, Harvard NOM Research Paper No. 03-60, Massachusetts, USA (2003)
- Gallivan, M.J.: Striking a balance between trust and control in a virtual organization: a content analysis of open source software case studies. *Information Systems Journal* 11(4), 277–304 (2001)
- Hinds, D., Lee, R.: Social Network Structure as Critical Success Condition for Virtual Communities. In: *Proceedings of the 41st Hawaiian International Conference on System Sciences* (2008) <http://csdl2.computer.org/comp/proceedings/hicss/2008/3075/00/30750323.pdf> (accessed January 6, 2009)
- Kilduff, M., Angelmar, R., Mehra, A.: Top management team diversity and firm performance: Examining the role of cognitions. *Organization Science* 11(1), 21–34 (2000)
- Knight, D., Pearce, C.L., Smith, K.G., Olian, J.D., Sims, H.P., Smith, K.A., Flood, P.: Top management team diversity, group process and strategic consensus. *Strategic Management Journal* 20(5), 445–465 (1999)
- Nakakoji, K., Yamamoto, Y., Nishinaka, Y., Kishida, K., Ye, Y.: Evolution patterns of open source software and communities. In: *Proceedings of the International Workshop on Principles of Software Engineering*, Florida, USA (2002)
- Page, S.E.: *The difference: How the power of diversity creates better groups, firms, schools and societies*. Princeton University Press, New Jersey (2007)
- Roberson, Q.M., Park, H.J.: Diversity reputation and leadership diversity as sources of competitive advantage in organizations. In: *Academy of Management Proceedings*, pp. F1–F6 (2004)
- Sampson, R.: R&D alliances and firm performance: The impact of technological diversity and alliance organization on innovation. *Academy of Management Journal* 50(2), 364–386 (2007)
- Schroder, H.M., Driver, M.J., Streufert, S.: *Human Information Processing*. Holt, Rinehart and Winston, New York (1967)

- Stewart, K.J., Ammeter, T.: An exploratory study of factors influencing the level of vitality and popularity of open source projects. In: Proceedings of the 23rd International Conference on Information Systems, Barcelona, Spain (2002)
- Tatli, A., Ozbilgin, M.: Managing diversity measuring success. Chartered Institute of Personnel and Development (2005),
http://resources.greatplacetowork.com/article/pdf/managing_diversity.pdf (accessed January 6, 2009)
- Tuomi, I.: Internet, innovation and open source: Actors in the network. *First Monday* 6(1) (2001), http://www.firstmonday.org/issues/issue6_1/tuomi
- von Krogh, G., Spaeth, S., Lakhani, K.R.: Community, joining and specialization in open source software innovation: a case study. *Research Policy* 32(7), 1217–1241 (2003)
- Weiss, D.: Measuring success of open source projects using web search engines. In: The First International Conference on Open Source Systems, Genova, Italy (2005)
- West, J., O'Mahony, S.: Contrasting community building in sponsored and community founded open source projects. In: Proceedings of the 38th Annual Hawaii International Conference on System Sciences, Hawaii (2005)
- Winston, M.D.: The importance of leadership diversity: The relationship between diversity and organizational success in the academic environment. *College and Research Libraries* 62(6), 517–526 (2001)

Web Search Based on Web Communities Feedback Data

Mehdi Adda¹, Rokia Missaoui², and Petko Valtchev³

¹ Département de Mathématiques, d'Informatique et de Génie,
Université du Québec À Rimouski
adda@ieee.org

² LARIM, Université du Québec en Outaouais
rokia.missaoui@uqo.ca

³ LATECE, Université du Québec À Montréal
valtchev.petko@uqam.ca

Abstract. The capability to easily find relevant information becomes increasingly difficult as the available content increases. Web Search Engines aim to assist users in finding pertinent information. To measure the relevance of a Web page (its rank), different strategies are used. However, page ranking is mainly conducted by relying on automatic assessment criteria. Hence, a gap is created between the effective relevance of a content and the computed one.

To reduce this gap, we introduce a framework for feedback-based web search engine development. To illustrate the effectiveness and the use of the proposed framework, we developed a web search engine prototype called *SocialSeeker*. Finally, we evaluated our approach from the end-user perspective and the results shown that feedback data can improve search engine results.

Keywords: Feedback data, search engine, web communities, social-based filtering.

1 Introduction

The World Wide Web is growing at a staggering rate. As a result, finding relevant information on the Web is often difficult. Web search engines such as Yahoo, Google and Live Search¹ help users to find more relevant information when conducting searches.

In order to collect information about web pages, search engines use robots [10,21]. The resulting information is then used to index the related Web pages. Based on those indexes, queries formulated by users are mapped to potentially relevant Web pages already crawled.

1.1 Web Page Ranking Strategies

Ranking search results is a fundamental preoccupation when designing Web search engines. The most widely used Web search engines such as Yahoo, Live and Google keep secret the algorithms used to rank Web pages. However, some elements that play an important role in the classification of Web pages are known [16].

¹ www.yahoo.com, www.google.com, www.live.com

Firstly, Web search engines take into account information contained in a Web page such as the title and the URL (*Universal Resource Locator*). Secondly, search engines use what is called *bots* or *spiders* to parse the structure of a Web page looking for information not always displayed by a browser and that is used to describe the content of a Web page, such as the words contained in the meta-keyword tags. Thirdly, the frequency and the context of the words in the Web page are taken into account.

However, those elements may be easily manipulated by Webmasters, which affects and biases the ranking of a Web page. To reduce this influence, some other factors less susceptible to be manipulated by Webmasters are integrated into the ranking process. It mainly consists of taking into account the amount of traffic generated by a Web page, as well as the number of websites that are referencing this page. Even with those extensions, classical search engines are completely dependent on an automatic classification process of Web pages. Thus, the computed rank may not correspond to the effective relevance of a Web page content. A solution to this problem is to exploit user's feedback in the ranking process. Such data may be found in social bookmarking Web applications. Concretely, this relevance may be measured using user evaluations of the content. In this paper, the expression *opinion data* is used to refer to these evaluations. The focus of the present work is to develop a framework to integrate opinion data into the Web search process and test its effectiveness.

1.2 Opinion Data and Web Search

Social bookmarking Web applications are rapidly emerging on the Web. In fact, more and more applications such as Dzone [4], Reddit [6], Digg [2], Slashdot [8], are developed in order to share content as well as collecting content evaluation by Web users [14, 18]. These applications can be seen as: (i) warehouses of links to Web resources, and (ii) an interface for a community-based evaluation and filtering of resources. The evaluations of users for the content can then be exploited to determine the "social/community" relevance of a resource [13].

Currently, there are few search engines that attempt to integrate opinion data in the ranking process of Web pages (eg. Mahalo [5], Dipiti [3], Chacha [11]). These systems allow users to evaluate the suggested content and then utilize this information to (re)evaluate the ranking of the related content. One limitation of these systems is the difficulty to integrate opinion data already available on the Web. Therefore, they generally require the creation of separate databases to collect and store opinion data.

Instead of creating a new database and starting to collect opinion data, we propose to exploit the already existing evaluations on a given web page to measure its relevance. Hence, we propose a framework where the relevance of a Web page is directly calculated from opinion data available in specialized social bookmarking sites. The framework we propose is mainly composed of a language to describe data and categories of opinions and of a set of operations to handle the different elements of the proposed language (cf. Section 2).

One of the difficulties we encountered is related to the diversity of the forms opinions are expressed on. For example, there are websites where only binary votes are accepted (appreciate, do not appreciate), other systems implement multi-scale votes (eg. level of assessment in a scale of 3, 5 or 10 levels).

Therefore, the main challenge is how to consolidate and deal effectively with heterogeneous feedback data. To resolve this problem and to compare relevance values obtained from different social bookmarking websites, we use a ranking formula that normalizes these values (*cf.* Section 2.4).

To show the effectiveness of the proposed framework and then evaluate the relevance of the output of this framework, we implemented a prototype called *SocialSeeker* [11] which is completely based on feedback data. The system seeks for Web pages related to a specific topic with the highest "social/community" evaluations (*cf.* Section 3). Whereas in section 3.1 we present the implementation, in sections 3.2 and 3.3 we show how we evaluated the proposed system and presented the results of those experimentations. Finally, we present in section 4 conclusive remarks and perspectives on improving the proposed framework and the developed prototype *SocialSeeker*.

2 Integrating Opinion Data in Web Search

Figure 1 represents an overview of the integration logic of feedback data into a Web search engine. In this figure, we present social networks as an example of applications for sharing references and resources. These applications, which will be referred to as *feedback warehouses*, gather explicit and/or implicit feedback data from Web users [19,20]. However, criteria and features offered by each application may differ. Therefore, it is necessary to analyze the provided feedback data to build an appropriate model for measuring the relevance of a Web content.

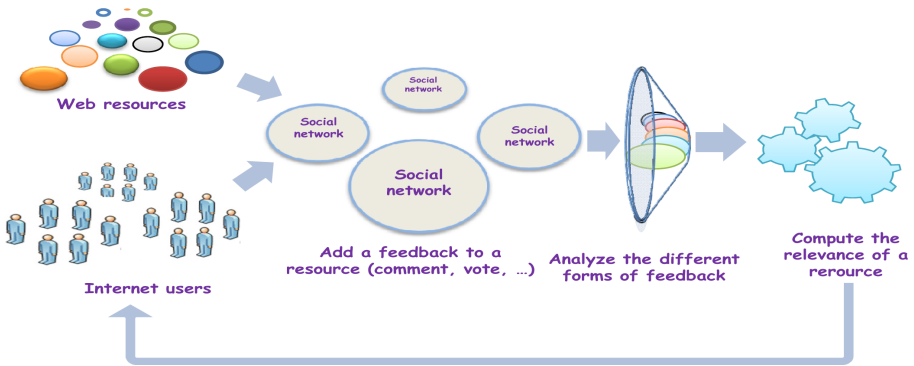


Fig. 1. Opinion data integration in Web search process

2.1 Typical Usage Scenario

To illustrate how a search engine that is based on feedback data operates, we present in Figure 2 a typical usage scenario. First, a user submits a query. Based on the keywords contained in this query, the system interrogates the available feedback warehouses. Thereafter, each warehouse returns a collection of references to Web resources

(eg. links, tags, summaries, ...) and the evaluations associated to each resource. The system retrieves the results and calculates a normalized relevance value for each element according to the evaluations associated to it. After classification, references with the greatest values are returned to the user. Finally, the user can leave an evaluation which will be directly forwarded to the related warehouse(s).

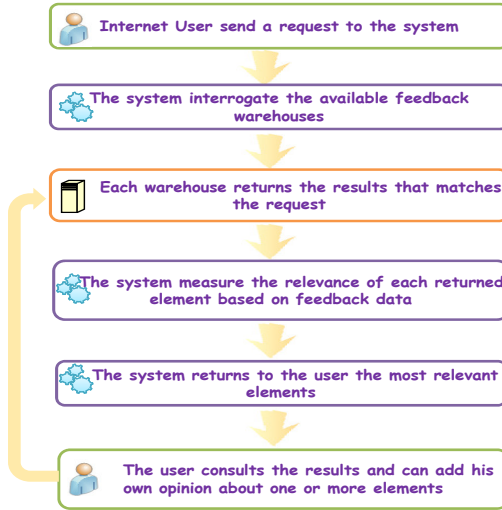


Fig. 2. Typical scenario of opinion based Web search

Opinion data are classified into categories known as feedback classes (cf. Section 2.2).

2.2 Opinion Data Description Language

In this section, we begin by formalizing the description of opinion data. Then, we propose a language based on description logics [12] to model the categories of such data. To distinguish between data aspects that are purely structural and the meaning of these structures, the language is endowed with a set of syntactical constructors to which are associated precise semantics. While the syntax defines the structure of data classes, the semantic part associates to them a meaning in the underlying domain (users' assessments).

Description of opinion data. The data we are handling are records, called *instances of opinion* (cf. Definition 2.4), which are composed of *references to Web resources* (cf. Definitions 2.1 2.2) and *feedback data* associated with these resources (cf. Definition 2.8). In the simplest configuration, a reference may be a simple *URL*, and the feedback data limited to the number of people who appreciated the content of the Web page indicated by this *URL*. A more complex configuration is to consider a reference as being composed of a set of elements in addition to the *URL*, such as a summary, tags,

the author’s name and the date of publication etc., and the feedback data that includes comments, positive votes, negative votes, and so on. The feedback data associated to a resource is represented by a set of couples (k, v) , where k refers to the feedback class (cf. Section 2.8), and v the associated value. It is noteworthy that the value associated to v may be numbers (votes, ...), or strings (keywords, comments, ...). A couple (k, v) is called *feedback object* and the universe of all couples (k, v) is denoted by Ψ . Table 1 presents some basic notions related to the present work.

Table 1. Basic notions

Symbol/Relation	Represented notion
\mathcal{R}	Set of Web resources i
$URLS_i$	Set of elements indicating the localisation(s) of the resource i in the Web. $\bigcup_{i \in \mathcal{R}} URLS_i = URLS$
ϖ_i	Content of the resource i . $\bigcup_{i \in \mathcal{R}} \varpi_i = \varpi$
χ_i	Information describing/completing a resource i . $\bigcup_{i \in \mathcal{R}} \chi_i = \chi$
Ψ_i	Set of feedback data associated with i . $\bigcup_{i \in \mathcal{R}} \Psi_i = \Psi$

A resource accessible via the Web is represented by the following elements: a Web address to localize the resource, the content of the resource, and possibly information describing the resource (cf. Definition 2.1). The set of all Web resources is denoted by \mathcal{R} .

Definition 2.1. Web resource

A Web resource i ($i \in \mathcal{R}$) is defined by $\langle url, ct, \mathcal{I} \rangle$, where:

- $url \in URLS_i$;
- $ct = \varpi_i$;
- $\mathcal{I} \subseteq \chi_i$.

Generally speaking, feedback-based social websites are referencing Web resources which are published in other websites. Moreover, a *reference to a Web resource* is composed of a set of information that represents the resource. In the following, we rely on the concept of Web resource to give a formal definition of a *reference to a Web resource* (cf. Definition 2.2).

Definition 2.2. Reference to a Web resource. Let $i \in \mathcal{R}$, and ρ a relation from the set of resources \mathcal{R} to the set of triplets $(URLS \times \varpi \times \chi)$. The set of references to the resource i is represented by $\rho(i) = \{e = \langle url, ct, \mathcal{I} \rangle | url = i.url \wedge ct \subseteq i.ct \wedge \mathcal{I} \subseteq i.\mathcal{I}\}$.

Proposition 2.3. If $i \in \mathcal{R}$, then $i \in \rho(i)$, which means, a Web resource is considered as a reference to itself.

Proof 2.1. Let i an element of \mathcal{R} ($i \in \mathcal{R}$). From Definition 2.2 a resource $e = \langle url, ct, \mathcal{I} \rangle$ belongs to the set of i references ($e \in \rho(i)$) if and only if $e.url = i.url$, $e.ct \subseteq i.ct$ and $e.\mathcal{I} \subseteq i.\mathcal{I}$. Because, $i.url = i.url$, $i.ct \subseteq i.ct$ and $i.\mathcal{I} \subseteq i.\mathcal{I}$ then $i \in \rho(i)$. We conclude that each resource is a reference to itself.

In community-based bookmarking websites, usually users provide assessments to a resource whose references are published on the same website, whereas the content itself is published in other websites. These assessments are represented in what we call *instances of opinion* and are composed of two distinct parts. An identification part, which is used to identify the considered resource, and a data part which represents the feedback data associated with this resource. Formally, an instance of opinion is defined as follows.

Definition 2.4. Instance of opinion. *An instance of opinion of the resource i is a couple (ς, θ) , where ς is a reference to i , and θ a list of feedback data attached to i . In other words, we have the following:*

- $\varsigma \in \rho(i)$;
- $\forall j \in [1..|\theta|] : \theta[j] \in \Psi_i$.

Remarks 2.5

1. For more concision, a feedback data record of an instance of opinion $o = (\varsigma, \theta)$ at the position j in θ will be referenced by $o.\theta[j]$, and the cardinality of θ will be referenced by $|o.\theta|$. Moreover, the universe of all instances of opinion will be represented by U_O .
2. In order to make a clear distinction between elements of $o.\theta$, we adopted the following notation: $o.\theta^n$ to reference entries whose values are numerical, and $o.\theta^t$ to reference entries with textual values.

Description language for feedback classes

Syntax. A feedback class represents a category of feedback objects of the same type (cf. Definition 2.6).

Definition 2.6. The syntax of a feedback class. *A feedback class c , whose universe is denoted by U_Ψ , is defined by the quadruplet $(T, D, Cst, OptimVal)$ where T represents the feedback type of c , D the domain of c , Cst a set of constraints on this feedback class, and $OptimVal$ is the optimal value associated with c .*

For example, the quadruplet $(PositiveRating, Integers, \{\geq 0, \leq 10\}, 10)$ is a feedback class that represents the category of positive votes. Every vote is an integer that is greater than or equal to 0 and less than or equal to 10. The optimal value of this class is set to 10.

Remarks 2.7. *Within this paper the calculation of optimal values of a class $c \in U_\Psi$ is achieved as follows:*

1. If the domain of c is numeric then $c.OptimVal$ is either the minimum value or the maximum value that satisfies all constraints in $c.Cst$ w.r.t $c.D$;
2. If the domain of c is textual then $c.OptimVal$ is either the empty set (\emptyset) or the infinity (∞).

Semantic of a feedback class. A feedback class is interpreted by a set of feedback objects. The interpretation is based on two elements: (i) the interpretation domain which consists of all feedback objects, and (ii) the interpretation function, with associate for each class a set of elements within the interpretation domain. Furthermore, a feedback object is linked to a class by the instantiation relationship defined below.

Definition 2.8. Instantiation relationship

Let $c \in U_{\Psi}$ and $d = (k, v)$ a feedback object where k is a feedback type and v the associated value ($d \in \Psi$). The object d is instance of c , denoted $d < c$, if and only if:

- $d.k = c.T$;
- $d.v \in c.D$;
- $\forall t \in c.Cst, t(d.v) = true$, ie. $d.v$ satisfies the constraint t .

For instance, the object (*PositiveRating*, 3) is an instance of the feedback class defined by (*PositiveRating*, *Integers*, $\{\geq 0, \leq 10\}$, 10). However, the feedback object (*PositiveRating*, 15) is not an instance of this class, because the constraint " ≤ 10 " is not satisfied.

Remark 2.9. The notation $[o]_c$ is used to designate the class of the feedback object o .

Feedback class subsumption. Feedback class subsumption is a partial order which orders a set of classes of different websites into hierarchies. Intuitively, two feedback classes are linked to each other with subsumption relationship if one of them is either a direct manifestation of the other class or has a more specific domain and a set of constraints that covers a larger domain. More formally, subsumption relationship on feedback classes is defined as follows.

Definition 2.10. Subsumption of feedback classes. Let c_1 and c_2 two classes from U_{Ψ} , we say c_2 subsumes c_1 , noted by $c_1 \sqsubseteq_c c_2$, if and only if:

1. $c_1.T = c_2.T$;
2. $c_1.OptimVal \leq c_2.OptimVal$;
3. $((\bigcap_{cst_i \in c_1.Cst} |D|_{cst_i}) \cap c_1.D) \subseteq ((\bigcap_{cst_j \in c_2.Cst} |D|_{cst_j}) \cap c_2.D)$, where $|D|_{cst_i}$ and $|D|_{cst_j}$ are the restrictions of D by the constraints cst_i and cst_j respectively.

The last condition ensures that the restriction of the domain c_1 by its constraints is covered by the restriction of the domain of c_2 with the constraints of the latter. Because it is possible to have $c_1.D \subseteq c_2.D$ and at the same time have the constraints of c_2 more restrictive than those of c_1 in such a way that the effective domain of c_1 become broader than the effective domain of c_2 .

For example, if $c_2 = (PositiveRating, [0..10], \{\}, 10)$, $c_1 = (PositiveRating, Integers, \{\geq 1, \leq 5\}, 10)$, then $c_1 \sqsubseteq_c c_2$.

It is noteworthy that feedback classes of a given community-based website are not comparable with each other.

Before explaining the method we propose to estimate the pertinence a Web resource, we introduce the concept of *warehouse of opinions*. The objective behind the introduction of this concept is to bring together the various concepts presented in the previous sections (feedback classes, instances of opinion,...) in one entity and thus distinguish between elements from different community-based websites.

2.3 Feedback Warehouses

In our model of domain representation, a feedback warehouse is defined by a set of feedback classes and a set of instances of opinion such as described below.

Definition 2.11. Feedback warehouse. Let U_Ψ and U_O be respectively the universe of feedback classes and universe of instances of opinion. A warehouse e_i of the set of warehouses, designed by \mathcal{E} , is a couple (s_c, s_o) such that :

- $s_c \subseteq U_\Psi$;
- $s_o \subseteq U_O$;
- $\forall o_i \in s_o, \exists c_j \in s_c$ s.t $o_i \prec c_j$.

A general view of feedback warehouses and their components are depicted in Figure 3.

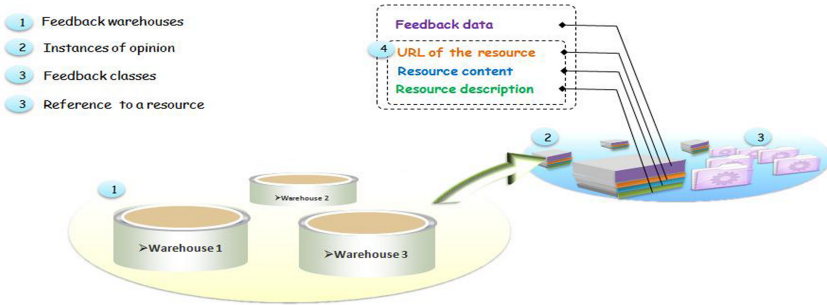


Fig. 3. Representation of feedback warehouses components

To find out if a warehouse is more general than another one, we introduce the warehouse subsumption relationship. Moreover, for each warehouse of opinions we define two particular instances. It consists of the warehouse *top* and the warehouse *bottom*. Therefore, these specific instances are later on used to measure the relevance of instances of opinion, which is essential in the ordering process of those instances. A naive approach will be to look for extreme instances in each warehouse and then measure the distance existing between each instance of the warehouse with these extremes. However, the search for these instances is a computationally expensive task because it requires to scan each warehouse twice: the first scan aims to find the extremes and in the second scan distances between each instance and these extremes are measured.

The second problem that this raises is the variability aspect of these extremes. In fact, an instance that is extreme at instant t may not remain extreme at instant t' (after the

insertion of new instances into the considered warehouse). To maintain the consistency of the system, after each insertion of an instance of opinion, we have to verify whether the extreme instances already selected still retain their status, if not replace them with the new extremes.

It is noteworthy that this method requires constant monitoring of warehouse updates inducing more resource consumption.

As a solution to those two problems, we propose to replace the effective extreme instances by virtual ones. These are based on the the optimal values of the warehouse feedback classes and guarantee the non existence (present and future) of other instances in the warehouse with more extreme values.

The advantage of these virtual instances is twofold: (1) they are calculated without scanning warehouses and (2) there is no need to constantly check each insertion to the warehouse.

Feedback Warehouses Subsumption. Feedback Warehouses subsumption relationship is a partial order that allows us to organize a set of warehouses depending on their feedback classes. This relationship is reduced to set inclusion of feedback classes. Formally, we have the following.

Definition 2.12. Warehouse subsumption. Let e_1 and e_2 two elements of the warehouses set \mathcal{E} , e_1 is more specific than e_2 , or e_1 is subsumed by e_2 , denoted $e_1 \sqsubseteq_e e_2$, if and only if $e_1.s_c \subseteq e_2.s_c$.

The top and bottom of a feedback warehouse. The top of a warehouse is an instance of opinion to which the optimal values ($OptimVal$) of the corresponding classes are associated. Formally, the top is defined as follows.

Definition 2.13. The top of a feedback warehouse. Let e an element of \mathcal{E} and $o \in U_O$, the instance o is considered as the top of e , denoted \hat{e} , if and only if:

1. $\forall k \in e.s_c : \exists j \in o.\theta$ s.t. $[j]_c = k$;
2. $\forall j \in o.\theta : \exists k \in e.s_o$ s.t. $[j]_c = k$;
3. $\forall j \in o.\theta : j.v = [j]_c.OptimVal$;

In the same way, the bottom of a warehouse is an instance of opinion to which are associated values that are *dual* to the optimal values of the warehouse classes (cf. Definition 2.14).

Definition 2.14. Bottom of a feedback warehouse. Let e an element of \mathcal{E} and $o \in U_O$, the instance o is a bottom of e , denoted by \underline{e} , if and only if :

1. $\forall k \in e.s_c : \exists j \in o.\theta$ s.t $[j]_c = k$;
2. $\forall j \in o.\theta : \exists k \in e.s_o$ s.t $[j]_c = k$;
3. $\forall j \in o.\theta : j.v = \{Max([j]_c.D \cup (\bigcap_{cst_k \in ([j]_c.Cst)} |D|_{cst_k})), Min([j]_c.D \cup (\bigcap_{cst_k \in ([j]_c.Cst)} |D|_{cst_k}))\} - [j]_c.OptimVal$;

Distance between two instances of opinion. Instances of opinion entries may be numerics (eg. a value from a set of 2, 3 or 10 numbers) or text (eg. comments, tags, ...). To take into account this fact when measuring the distance between two instances of opinion, we present herein a formula composed of two measures: one to compute the distance between numerical entries and the second to compute the distance between textual entries.

In order to compute the distance between numerical values, we use the *Minkowski* formula of order p [23]. The order p represent the number of feedback classes of each warehouse. The distance between text entries of two instances of opinion, is based on the portion of words that are present in one entry and not in the other. Based on those considerations, we propose a general formula that measures the distance between two instances (cf. Definition 2.15).

Definition 2.15. Distance between two instances of opinion. Let $e \in \mathcal{E}$ and o_1, o_2 two instances of opinion in the feedback warehouse e ($o_1 \in e.s_o$ and $o_2 \in e.s_o$). The distance between o_1 and o_2 , denoted by $d_e(o_1, o_2)$, is as follows: $d_e(o_1, o_2) = d_e^n(o_1, o_2) + d_e^t(o_1, o_2)$ where:

- $d_e^n(o_1, o_2)$ which represents the Minkowski distance between numerical entries of o_1 and o_2 , is computed as follows:

$$d_e^n(o_1, o_2) = \sqrt[p]{\sum_{j \in [1..|o_1.\theta^n|]} (o_1.\theta^n[j].v - o_2.\theta^n[j].v)^{|e.s_c|}}$$

- $d_e^t(o_1, o_2)$ which represents the distance between textual entries of o_1 and o_2 is given by the following formula:

$$d_e^t(o_1, o_2) = \begin{cases} 0, & |\alpha| \in \{0, \infty\} \text{ et } |\beta| = |\alpha|; \\ 1, & (|\alpha| \in \{0, \infty\} \text{ et } |\beta| \neq |\alpha|) \text{ or} \\ & (|\beta| \in \{0, \infty\} \text{ et } |\beta| \neq |\alpha|); \\ \frac{\frac{|\alpha-\beta|}{|\alpha|} + \frac{|\beta-\alpha|}{|\beta|}}{2}, & \text{else.} \end{cases}$$

Where $\alpha = \bigcup_{j \in [1..|\theta|]} o_1.\theta^t[j].v \cup o_1.\varsigma.ct \cup o_1.\varsigma.\mathcal{I}$, $\beta = \bigcup_{j \in [1..|\theta|]} o_2.\theta^t[j].v \cup o_2.\varsigma.ct \cup o_2.\varsigma.\mathcal{I}$ and $|text|$ represents the number of words comprised in text.

2.4 Measuring the Relevance of an Instance of Opinion and Querying a Set of Warehouses of Opinions

The relevance of an instance of opinion o in the warehouse e , denoted by $\mathcal{P}_e(o)$, is the distance between o and the bottom of e , ie. $\mathcal{P}_e(o) = d_e(o, \underline{e})$.

The values returned by \mathcal{P}_e are not normalized. However, in order to classify two instances belonging to two different warehouses, it is necessary to have normalized measure that generates normalized values. To this end, we propose the function $\overline{\mathcal{P}_e}$ which yields normalized values in the interval $[0..1]$. This function, is defined as follows.

Definition 2.16. Normalized relevance. Let $e \in \mathcal{E}$, $o \in e$, and $x = \mathcal{P}_e(o)$. The normalized relevance of o in e is given by $\overline{\mathcal{P}}_e(x)$ such that $\overline{\mathcal{P}}_e$ is defined from U_O to $[0..1]$ as follows:

$$\overline{\mathcal{P}}_e(o) = \begin{cases} 0, & \text{if } \mathcal{P}_e(\hat{e}) = 0; \\ |\frac{\mathcal{P}_e(o)}{\mathcal{P}_e(\hat{e})}|, & \text{otherwise.} \end{cases}$$

It is noteworthy that the maximum relevance value of an instance of opinion is reached when each associated feedback object have a value equal to the optimal value of the corresponding feedback class.

Now that the function $\overline{\mathcal{P}}_e$ computes normalized relevance values, resources described by instances of opinion of different warehouses can be compared. Hence, resources are classified in a descendant normalized relevance order and the first k elements are sent to the user (cf. Definition 2.17).

Definition 2.17. K-results. Let \mathcal{E}_1 be a set of warehouses ($\mathcal{E}_1 \subseteq \mathcal{E}$), Q a set of keywords, k an integer. The result of querying \mathcal{E}_1 with Q is a set of instances of opinion that belong to elements of \mathcal{E}_1 . Those instances are computed by the function $\mathcal{F}_{\mathcal{E}_1}$, such that an instance of opinion $o_i \in \mathcal{F}_{\mathcal{E}_1}(Q, k)$ if and only if:

- $\exists e \in \mathcal{E}_1$ s.t. $o_i \in e$;
- $\exists t \in Q$ s.t. $t \in (o_i.\varsigma.ct \cup o_i.\varsigma.\mathcal{I})$;
- $k' \leq k$ where $k' = |\{o_j \in U_o | (\exists e \in \mathcal{E}_1$ s.t. $o_j \in e) \wedge (\exists t \in Q$ s.t. $t \in (o_j.\varsigma.ct \cup o_j.\varsigma.\mathcal{I})) \wedge (\frac{\sum_{e \in \mathcal{E}_1} \overline{\mathcal{P}}_e(o_j)}{|\mathcal{E}_1|} > \frac{\sum_{e \in \mathcal{E}_1} \overline{\mathcal{P}}_e(o_i)}{|\mathcal{E}_1|})\}|$.

3 Evaluation

The goal of our experimentation is to study how relevant are the results returned by a search engine based on feedback data compared to the results that are obtained using classical search engines.

In order to fulfill this goal, we went through two phases. The first phase consisted on the implementation of *SocialSeeker*, a search engine prototype (cf. Section 3.1) based on the framework we have proposed in Section 2. In the second phase, we used the resulting implementation to evaluate our approach.

As the relevance of a result depends on the user viewpoint and judgement, we conducted an empirical evaluation which more suitable for our study (cf. Sub-section 1 of Section 3.2). In other words, the study has focused on comparing the scores that users associated to the results of requests that had targeted both *SocialSeeker* and a classical search engine (in our case it was *Google*).

3.1 Prototype

SocialSeeker is the name of the search engine prototype we have implemented. The primary objective of its development is to demonstrate the feasibility of the our approach. This engine has been implemented using *Rails 2.02* [17], an MVC-based (Model-View-Controller) Ruby [7] web development framework. Thus, the code is written mainly



Fig. 4. *SocialSeeker* home page

using *Ruby*, a dynamic programming language, and *HTML/CSS* for programming the graphical user interfaces. The prototype has been deployed in test mode using an application server called *Thin* [15] and *Nginx* [22] as frontend server. To date, *SocialSeeker* exploits two warehouses of opinions : *Dzone* and *Reddit*. In order to crawl those sites, we use *hpricot* [24] library written in *Ruby*. Figure 4 is a screenshot of the home page of *SocialSeeker*.

3.2 Experimentation Protocol

To ensure a maximum of objectivity in our study, we set an experimental protocol. The different steps of this protocol are as follows:

1. Provide the user with a unified interface so that he/she can submit keyword queries;
2. Retrieve the search query of the user;
3. Run this query on *SocialSeeker*;
4. Run this query on *Google*;
5. Assemble the top five results in the same list (10 items) by alternating between results of the two search engines;
6. Return anonymous results to the user (without indicating which search engine returned a given element);
7. Ask the user to specify the result(s) that are the most relevant to his/her request;
8. Retrieve and analyze the related data.

Analyse user evaluations is a relatively simple task. It mainly consists at measuring user satisfaction levels with the results given by the two search engines. Thus, in our study, the relevance of a query results is measured by the level of satisfaction of the user who submitted the request. Our first experiments have been conducted with 9 subjects. Given that *SocialSeeker* uses feedback sources mainly focused on new technologies then other fields, we restricted the comparison to the covered fields.

Each user has conducted a number of requests and evaluated part of result elements returned by *SocialSeeker* and *Google*. It is noteworthy that the system adds to each element of the result two options: appreciates and do not appreciate, from which the user have to choose one.

The results of the study are presented in Table 2. While the first line shows the number of requests made by each user, the second line shows the number of results that have been evaluated (positively or negatively) by each user.

Table 2. Number of requests and number of result elements evaluated by each user

	U_1	U_2	U_3	U_4	U_5	U_6	U_7	U_8	U_9
Requests	15	8	12	30	16	7	24	12	4
Evaluated elements	3	22	16	72	44	9	69	43	15
Results of <i>SocialSeeker</i>	2	13	4	10	37	5	24	20	14
Results of <i>Google</i>	1	9	12	62	7	4	35	23	1

From Table 2, we calculated the proportion of positive evaluations for each user for each search engine. The results are depicted in 3.

Table 3. Proportion of result elements positively evaluated for each search engine

	U_1	U_2	U_3	U_4	U_5	U_6	U_7	U_8	U_9
SocialSeeker									
Positively evaluated	100%	76,92%	75%	90%	89%	100%	29%	45%	71,42%
Google									
Positively evaluated	100%	100%	33,33%	22,58%	42,85%	75%	91,42%	95%	100%

Table 4 presents two measures related to data of Table 3. The first column shows the average proportion of satisfied users per search engine. Note that we consider a user as satisfied with the results of a search engine, if the percentage of its positive evaluations is over 50%.

We also calculated the average proportions of positive evaluations for different users (cf. column 2 of Table 4).

Table 4. Proportion of results evaluated per user per search engine

	Proportion of satisfied users	Proportion of satisfaction
SocialSeeker	77,77%	75,14%
Google	66,66%	73,35%

3.3 Results Analysis

As shown in Table 4, the proportion of users satisfied with the results returned by *SocialSeeker* is higher than the proportion of users satisfied with the results of *Google*. From the same table, one can see that *SocialSeeker* offers slightly a higher satisfaction average than *Google*. One of the main reasons that we believe to be behind the improvement, is related to the nature of data sources *SocialSeeker* is using. In fact, it is based on specialized community sites and the reputation of the original source does not bias the quality of results. For example, using *java* as a search keyword, the first result elements returned by *Google* are related to *Sun* [9] (or affiliated sites), while with *SocialSeeker* you find links to articles and resources that are "up-to-date" (techniques and paradigms, book reviews, new trends, specialized articles, blog entries, ...).

From those results, the first pattern that emerges is as follows: in the classical search engines, the relevance of a Web page seems to be biased by the size/reputation of the sites of these pages. The second pattern is that Web users seems to be more interested in content that is not necessarily published by/on big companies websites, but rather interested in content than their peers have evaluated positively.

The results of these initial experiments are encouraging. However, to draw definitive conclusions about the relevance of feedback-based web search, large-scale experiments are needed. These should cover a larger user base and a broader spectrum of Web communities. To achieve this goal, the next step is to turn *SocialSeeker* from an experimental prototype to a production ready system. This will not only enable us to confirm/invalidate the results achieved by these initial experiments but will also allow us to test the scalability of the proposed model in an environment where thousands if not millions of requests will hit the system.

4 Concluding Remarks and Perspectives

In this paper we have proposed a framework for integrating feedback data into search engines. It is mainly composed of a data description language and a set of operations on the elements of this language. This framework takes into account the heterogeneity of data and allows us to compare feedback data of different sources by using a standardized measure for calculating the relevance. The latter takes into account the positive and negative assessments of web users.

The proposed model is then used to implement *SocialSeeker*, a prototype search engine which uses the feedback data available on the Web for resource classification and filtering. Based on this implementation, we evaluated the proposed approach and presented the results of our experimentation.

It is noteworthy that integrating feedback data is not without risk. Indeed, it is likely that the calculation of the relevance of a resource may be biased by some factors such as the degree of expertise of evaluators, their subjectivity and the context in which feedback data are gathered. Further studies are necessary to not only qualify the potential bias but also to quantify it. With regards to the implementation, *SocialSeeker* relies exclusively on a manual population procedure of its warehouses database. For more flexibility, we plan to make available a public API of the system. This will ease the task of editing the warehouses base programmatically.

In order to push the validation process further, and test the scalability of the approach, it will be very interesting to expand *SocialSeeker* from a *proof-of-concept* to a production mode. Finally, we believe that classical search engines may benefit from this work to improve the relevance of their results by integrating feedback data in page ranking.

References

1. Chacha - mobile search — text search — questions and answers, <http://www.chacha.com>
2. Digg - all news, videos, & images, <http://www.digg.com>

3. Dipiti. funny name for human-filtered search, <http://www.dipiti.com>
4. Dzone.com - fresh links for developers, <http://www.dzone.com>
5. Mahalo.com: Human-powered search, <http://www.mahalo.com>
6. Reddit.com: what's new online!, <http://www.reddit.com>
7. Ruby programming language, <http://www.ruby-lang.org/en>
8. Slashdot - news for nerds, stuff that matters, <http://www.slashdot.org>
9. Sun microsystems, inc., <http://java.sun.com/>
10. Adah, S., Bufi, C., Temtanapat, Y.: Integrated search engine. In: KDEX 1997: Proceedings of the IEEE Knowledge and Data Engineering Exchange Workshop, pp. 140–147 (1997)
11. Adda, M.: Socialseeker: community-based search engine (2008), <http://socialseeker.no-ip.org/>
12. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.: The description logic handbook: Theory, implementation and applications, p. 574. Cambridge University Press, Cambridge (2003)
13. Bojrs, U., Breslin, J.G., Finn, A., Decker, S.: Using the semantic web for linking and reusing data across web 2.0 communities. *Web Semant.* 6(1), 21–28 (2008)
14. Borgatti, S.-P., Cross, R.: A relational view of information seeking and learning in social networks. *Manage. Sci.* 49(4), 432–445 (2003)
15. Cournoyer, M.-A.: Thin: open-source, ruby web server (2008), <http://code.macournoyer.com/thin/>
16. Ding, C., Chi, C.-H.: A generalized site ranking model for web ir, 584–587 (2003)
17. Hansson, D.H.: Ruby on rails: open-source web framework (2008), <http://www.rubyonrails.org>
18. Hooff, B.v.d., Elving, W., Meeuwssen, J.M., Dumoulin, C.: Knowledge sharing in knowledge communities. *Communities and technologies*, 119–141 (2003)
19. Joachims, T., Granka, L., Pan, B., Hembrooke, H., Radlinski, F., Gay, G.: Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM Trans. Inf. Syst.* 25(2), 7 (2007)
20. Shahabi, C., Chen, Y.-S.: An adaptive recommendation system without explicit acquisition of user relevance feedback. *Distrib. Parallel Databases* 14(2), 173–192 (2003)
21. Sun, J., Zeng, H., Liu, H., Lu, Y., Chen, Z.: Cubesvd: A novel approach to personalized web search. 14th international conference on World Wide Web, 574 (2005)
22. Sysoev, I.: Nginx: open-source, high-performance http server and reverse proxy (2008), <http://wiki.codemongers.com/>
23. Varadhan, G., Manocha, D.: Accurate minkowski sum approximation of polyhedral models. *Graph. Models* 68(4), 343–355 (2006)
24. Why, Hpricot: flexible html/xml parser (2008), <http://code.whytheluckystiff.net/hpricot/>

Improving Trust and Reputation Modeling in E-Commerce Using Agent Lifetime and Transaction Count

Catherine Cormier and Thomas T. Tran

School of Information Technology and Engineering
University of Ottawa
Ottawa, Ontario, Canada K1N 6N5

Abstract. Effective and reliable trust and reputation modeling systems are central to the success of decentralized e-commerce systems where autonomous agents are relied upon to conduct commercial transactions. However, the subjective and social-based qualities that are inherent to trust and reputation introduce many complexities into the development of a reliable model. Existing research has successfully demonstrated how trust systems can be decentralized and has illustrated the importance of sharing trust information, or rather, modeling reputation. Still, few models have provided a solution for developing an initial set of advisors from whom to solicit reputation rankings, or have taken into account all of the social criteria used to determine trustworthiness. To meet these objectives, we propose the use of two new parameters in trust and reputation modeling: agent lifetime and total transaction count. We describe a model that employs these parameters to calculate an agent's seniority, then apply this information when selecting agents for soliciting and ranking reputation information. Experiments using this model are described. The results are then presented and discussed to evaluate the effect of using these parameters in reputation modeling. We also discuss the value of our particular model in contrast with related work and conclude with directions for future research.

Keywords: Trust, Reputation, E-commerce, Multi-Agent Systems, Agent Lifetime, and Transaction Count.

1 Introduction

In any multi-agent system, autonomous agents must be able to determine which other agents it trusts for any given interaction. "Trust is central to all transactions" as stated by Dasgupta [4]. Since e-commerce systems exist to facilitate transactions, it follows that trust is key to all e-commerce environments.

In order to have a successful e-commerce system, it is imperative that reliable and effective trust models be in place. The critical challenge in developing a sound trust model is that trust is subjective [6]. As well, the open and distributed nature of multi-agent e-commerce systems where agents act autonomously, based on their own interests, values and beliefs makes designing a robust trust model difficult. Given the

significance and complexity of the problem, a number of researchers have tried to tackle this problem and have proposed systems that assess trust using different methods and parameters [3, 7, 8, 10, 11, 12, 13].

Since “trust is based on reputation” [4] many of these approaches consider agent reputation a key factor in the trust model. Reputation is based on past behavior observed and reported by (e.g., via word-of-mouth) other agents and is typically communicated between agents using a reputation rating. Reputation modeling is the design of approaches to (i) generate, (ii) discover and (iii) aggregate rating information [13]. This paper aims to recommend new parameters that can be used to enhance reputation modeling.

Several successful online marketplaces such as eBay and the Apple App Store offer centralized systems for reporting rating information [1, 2]. At the eBay site, other users (buyers and sellers) are rated, while at the Apple App Store, software applications available for purchase are rated. In both of these systems, ratings are generated by users once they have engaged in a transaction. The ratings are published publicly at the site for discovery by other users. Rating aggregation is then left to the individual user; each may interpret the ratings and other information about the user or application and make a trust decision in their own manner.

In these online marketplaces, the fact that a user must complete a transaction prior to submitting a rating suggests that a user’s opinion is valued only if it is based in experience. Moreover, the influence that a user’s opinion may have increases with the number of transactions in which the user participates (because the user’s opinion may appear up to once per transaction). From these observations, we assert that the value of a reputation rating is related to the number of transactions in which an agent has participated.

In addition, eBay presents a summary for each user, which includes a “member since” date as well as a total number of ratings received. As described above, the number of ratings must be less than or equal to the number of transactions in which the user has participated, and is therefore closely related to the number of transactions in which the subject has participated. Since both the membership date and number of ratings are readily available in the description of all sellers, as well as the users who have rated the seller we suggest that this information must influence the buyer’s decision to trust the seller and its aggregation of ratings.

In our research, we have sought to investigate the effectiveness of applying these principles observed in successful centralized e-commerce systems in decentralized multi-agent e-commerce systems. We propose that reputation modeling in distributed e-commerce systems can be improved by considering the amount of time an agent has been part of a system, or *agent lifetime*, and the number of transactions in which an agent has participated, or *transaction count*, when discovering and aggregating reputation ratings.

We present our research and findings in this paper as follows: In section 2 we discuss related work. We then formally present the proposed parameters and approach in section 3. In section 4 we describe our experimental technique and results. A discussion is given in section 5 and we conclude with future work in section 6.

2 Related Work

Due to the importance of having reliable and effective trust and reputation systems in e-commerce marketplaces, trust and reputation modeling has become an active area of research. As a result, over the past several years a variety of proposed approaches have emerged.

Many of these models propose using a *direct trust* rating in conjunction with *reputation ratings* obtained from a set of advisor agents [1, 3, 7, 8, 10, 11, 12, 13]. Generally, *direct trust* is the trust that one agent has in another based on its past experiences, for example, the trust that agent *a* has in agent *b*. *Reputation ratings* represent the reputation information supplied by an advisor agent, for example, the rating that agent *c* reports to agent *a* about agent *b*.

Some of the models also consider additional facets of trust. For example, REGRET combines the individual, social and ontological dimensions of trust [12]. In this case, not only direct trust and reputation ratings are considered, but also information about the agent's group and ratings of different aspects of an agent's performance are factored. In FIRE, direct trust ("interaction trust"), role-based trust (determined from pre-set rules about ratings for different relationships), reputation ratings ("witness reputation") and certified reputations (provided by the agent itself but signed by a recommender) are combined to determine an aggregated trust value [8].

A few of the models are experience or evidenced-based, in particular REGRET [12], FIRE [8], CertainTrust [11], as well as Hang et al.'s evidence-based model [7] and Reece et al.'s experienced-based model [10]. In these approaches, the number of experiences or pieces of evidence on which a rating is based is considered in the computation of the aggregated rating. This is important, as claimed by Hang et al., because when a rating estimated by a single value, such as a probability, it is impossible to know whether that rating is based on very few or many experiences [7]. And, as Ries points out, "the certainty of an opinion increases with the number of evidence on which that opinion is based" [11]. Thus, these systems recognize and support the principle that advice based on greater experience is of more value. However, these systems consider only the number of interactions between the agent rating and the agent being rated; they do not consider an agent's experience in the system overall.

Finally few systems specifically define the process for initializing the list of advisor agents. In fact, Abdul and Hailes indicate that their model is not suitable for bootstrapping the advisor list [1]. Others, such as [7], [8], [10] and [13] simply use a set of neighbors as their advisor list.

As with others who have developed experience and evidence-based models, we propose that an opinion that is based on a high level of experience is generally more valuable than an opinion based on less experience. In contrast to these approaches, which consider strictly the number of experiences directly between two agents, however, we suggest considering an agent's overall experience level. This experience level, based on the two metrics agent lifetime and transaction count, can be used as a determining factor for trusting the advice an agent provides. We also use these parameters to develop an initial advisor agent list, which is a problem generally left unaddressed by the above approaches.

3 Proposed Approach

For the purposes of improving the initial selection of advisor agents and to accurately weight the advice received by advisor agents, we propose the use of agent lifetime and transaction count. The use of these parameters is derived from the principle that advice obtained from the most experienced members of a group is generally the most valued. It is proposed that by using these factors to select initial advisors, novice agents can benefit from the experience of more senior agents, thereby reducing risk by achieving desirable results sooner.

3.1 Agent Lifetime

Agent lifetime is the amount of time that an agent has been a part of the multi-agent system, or, more simply, the agent's age within the system. A characteristic of open distributed systems is the agent's ability to enter and leave the system freely. As Ramchurn et al. point out, this characteristic of the system can be leveraged by malicious agents who leave and reenter the system in order to change their identities and escape their past behavior [9]. Conversely, agents who have established a positive reputation over time would be better suited to stay within the system. As a result, an agent's lifetime can be used as an indicator of trustworthiness.

Agent lifetime is calculated at any given time t using a *timestamp* assigned to the agent when it enters the multi-agent system. For any agent a , with assigned timestamp T_a , the agent lifetime is given by:

$$l_a(t) = 1 - \frac{T_a}{t} \quad (1)$$

Using this formula, agent lifetime is normalized so that $l_a(t)$ is always in the range $[0,1]$. Agents that have been in the system for a long time will have a lifetime approaching 1, while agents that have recently entered the system will have a lifetime close to 0.

3.2 Transaction Count

An agent's *transaction count* is the total number of transactions in which the agent has participated over the course of its lifetime. This is a measure of the agent's activity level within the system. It is presumed that an agent that has participated in a large number of transactions is more experienced than an agent that has participated in fewer transactions. And, by extension, the advice provided by the agent with a higher transaction count is more valuable. Exceptions may exist where, for example, malicious agents engage in a large number of low-value transactions in order to falsely inflate their transaction count and to use ballot-stuffing techniques to falsely inflate or deflate another agent's reputation. Agent lifetime and transaction count could be used to detect such malicious behavior, for example by determining that an agent with a short lifetime has engaged in a suspiciously high number of transactions. This application of agent lifetime and transaction count is beyond the scope of the research presented here; therefore, we defer it to our future work (Section 6).

The transaction count for any agent a is denoted by n_a and is automatically incremented by the system whenever a transaction occurs between agent a and any other agent.

3.3 Seniority

In the proposed approach, an agent who has been an active participant of the multi-agent system for a relatively long time, as compared with the other agents in the system, is considered a senior agent within the system. By identifying agents who are the most senior in the system, agents entering the system will be able to establish an initial set of advisors whose recommendations are based on as much experience as possible. Further, agents' seniority can be used to weight the advice received from advisor agents so that advice based on more experience more heavily impacts the overall reputation calculated.

The *seniority* of agent a at any time t is given by the product of its lifetime and transaction count:

$$s_a(t) = n_a \cdot l_a(t) \quad (2)$$

An agent a is said to be more senior than agent b if $s_a(t) > s_b(t)$. Any agent a that is new to the system or has never participated in a transaction is considered a novice agent and has $s_a(t) = 0$.

3.4 Building an Initial Advisor List

By adding agent lifetime and transaction count to a reputation model, novice agents are able to build an informed initial advisor list and therefore make good selections of agents with whom to engage in transactions, even with no or limited experience of their own. In order to find N advisors upon entering the system, the agent collects a list C of all candidate advisors that it can discover—this could be a list of neighbors, referred advisors or agents discovered using some other technique. It then calculates the seniority $s_c(t)$ for each candidate advisor c in the set C . Finally, the agent selects the N candidate advisors with the highest $s_c(t)$ values as its advisor list, denoted as A .

As time progresses, agent a may wish to refresh its list of advisor agents by replacing one or more of the advisors in its list. At that time, the agent may use the same or a similar technique to select a new set of advisors by using seniority values calculated at that time.

3.5 Weighting Advice

Once an agent has established its list of advisors and is faced with the decision to participate in a transaction with another agent, it will solicit advice in the form of a reputation ranking from each of its advisors. Since advice provided by advisors who are more experienced is deemed more valuable than advice given by those who are less experienced, the seniority of the advisor agent can be used to weight the reputation rating received from each advisor.

In order to decide whether or not to trust agent b , agent a solicits advice about agent b from each advisor agent adv_i in its advisor list A . When each advisor agent

receives the request for advice, it can respond by sending its reputation rating r_{adv}^b to agent a . Agent a then computes a total reputation rating for agent b , $r_b(t)$, using each advisor's current seniority to weight the advice received, as follows:

$$r_b(t) = \frac{\sum_{adv_i \in A} r_{adv_i}^b \cdot s_{adv_i}(t)}{\sum_{adv_i \in A} s_{adv_i}(t)} \quad (3)$$

This reputation value can then be combined with the direct trust rating that agent a has for b (based on its own previous experiences with b , if any). This can be accomplished using a simple technique such as computing the average of the reputation value and direct trust rating, or more elaborately following the techniques such as those presented by other researchers [1, 3, 7, 8, 10, 11, 12, 13]. However, in order to specifically examine the effect of using this approach, this aggregated reputation value $r_b(t)$ is used alone for selecting agents in the experiments described in Section 4.

4 Experimental Results

We implemented a software simulation to examine whether the use of agent lifetime and transaction count as described in Section 3 would: (a) enable agents who are new to the system to make effective decisions immediately; (b) enable agents to make more effective decisions overall. To determine the effects of these parameters independently of other factors, we employed a very simple reputation modeling system as presented in Section 4.1.

Furthermore, for analysis purposes, both a base case model and a test case model were implemented, as described in detail below. The differences between the two models are kept as minimal as possible, thereby further isolating the effect of the agent lifetime and transaction count parameters.

4.1 Description

The test software simulates an e-commerce marketplace where buyer agents may purchase from any seller agents. To simplify the experiment, it is assumed that all selling and buying occurs in the same context. That is, all sellers are offering competitive products and all buyers are in the market to purchase similar products. Furthermore, all buying agents can act as advisor agents to other buying agents.

Advisor agents report their reputation rating, a value in the continuous range $[-1, +1]$, for a given seller at the request of a buying agent. As an advisor agent, the agent may be either: honest, dishonest_high, dishonest_low or dishonest_erratic. An honest agent truthfully reports the average of its internal ratings for that seller, where each internal rating is simply the average utility it has gained from a transaction with the seller, normalized to be in the $[-1, +1]$ range. Dishonest agents constantly report either a high, low or erratic value, depending on their advisor type. Specifically, the reputation rating that each advisor type provides is as follows:

Table 1. Reputation ratings returned by different advisor agent types

Advisor Type	Reputation Rating Returned
Honest	Average of internal ratings for seller in question
Dishonest_high	Random value in the range [0, +1.0]
Dishonest_low	Random value in the range [-1.0, 0]
Dishonest_erratic	Random value in the range [-1.0, +1.0]

When a new buying agent is created and enters the marketplace, it initializes its list of advisor agents using either the technique described in the base case model or the test case model, as described in sections 4.2 and 4.3, respectively.

On each time step, every buying agent in the marketplace is given the opportunity to buy. To simulate agents with a variety of transaction counts, each buying agent decides whether or not to buy based on its buying activity level, which can be constant, high, medium, low or very low. Buyers with a constant activity level must buy on every time step, while agents with a very low activity only buy on every 11th time step. Other buyers determine how many time steps to wait between purchases by randomly selecting a wait period over a given range, as specified in Table 2.

Table 2. Time steps between purchases for different buyer activity levels

Buyer Activity Level	Time steps between purchases
Constant	0
High	Random value in the range [0, 2]
Medium	Random value in the range [3, 6]
Low	Random value in the range [7, 9]
Very Low	10

Once a buying agent decides to buy, for each candidate seller s , the buying agent asks all of its advisors for their advice about s . If the buying agent does not receive any advice about s , it adds s to its list of *unrated sellers*. If it does receive reputation ratings for s , the buying agent aggregates all of the ratings received, calculating r_s by following either the base case model (Section 4.2) or the test case model (Section 4.3). It then compares this aggregated reputation rating against its personal *trust threshold*, which is the minimum reputation rating the selling agent must have to be selected. This value represents the buyer's preference and could therefore vary from one agent to another in practice. However, for this purpose of this simulation, every buying agent has the same trust threshold, as given in Table 4. If the aggregated reputation rating for the most highly rated seller is greater than the buyer's trust threshold, then the buyer proceeds with that seller. Otherwise, the agent randomly selects from any unrated sellers or, if no unrated sellers are available, selects the highest rated seller (even though its rating was below the *trust threshold*).

When a buyer chooses to buy from a particular seller, the buyer receives a utility value which is a discrete value in the range [0, 10], where 0 denotes a very bad outcome and 10 denotes a very good outcome. Sellers may be good, average, bad or erratic, which means that they will randomly return a value in the corresponding range, as specified in Table 3.

Table 3. Utility range for different types of selling agents

Seller Type	Utility Range
Good	[7, 10]
Average	[4, 6]
Bad	[0, 3]
Erratic	[0, 10]

After the transaction is complete, the buying agent converts the utility received to a trust rating in the continuous range $[-1,1]$ and stores it in its internal rating table for the corresponding seller.

For each run, the simulated marketplace is initialized with a set number of buyers, N_BUYERS and sellers $N_SELLERS$. In order to simulate agents with different lifetimes and transaction counts, a new group of buyers is added to the marketplace after each interval I of time steps has passed. The simulator continues to add groups of buyers to the marketplace every I time steps. These groups are each assigned a number so that their behavior may be analyzed as a group. As well, each buying agent refreshes its advisor list by creating a new list every J th time step.

4.2 Base Case Model

The base case model is a simple approach that is provided to compare and evaluate the test case model (i.e., the proposed model that employs the use of the agent lifetime and transaction count parameters). The base case model and test case model differ in two respects: (i) how buying agents select their list of advisor agents; (ii) how buying agents aggregate the advice received from advisor agents.

Selection of Advisor Agents: In the base case model, buying agents generate their list of advisor agents by randomly selecting $N_ADVISORS$ advisor agents from all of the possible advisor agents (i.e., all other buying agents) in the marketplace.

Calculating Seller Reputation: For the base case model, buying agents compute the average of all of the ratings received from advisor agents. Therefore, if buying agent b is evaluating selling agent s , it solicits advice from all agents in its advisor agent list then computes:

$$r_s = \frac{\sum_{r_i \in R} r_i}{|R|} \quad (4)$$

where r_s is the consolidated reputation rating and R is the set of reputation ratings obtained from the buying agent's advisors.

4.3 Test Case Model

The test case model employs the proposed parameters agent lifetime and transaction count, following the proposed approach presented in Section 3. It differs from the

base case only in how buying agents select advisor agents and in how it uses ratings provided by advisor agents to calculate seller reputation.

Selection of Advisor Agents: In the test case model, when a buying agent needs to select advisor agents, it first computes the seniority of all candidate advisor agents (i.e., all other buying agents in the marketplace) following equation (2). It then selects the $N_ADVISORS$ agents that have the highest seniority values as its list of advisors.

Calculating Seller Reputation: To aggregate the reputation ratings received from its advisor agents, each buying agent in the test case model uses the seniority weighting formula given in equation (3).

4.4 Results

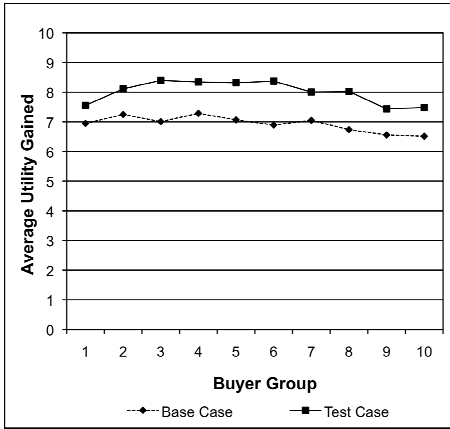
The simulation marketplace was run ten times using each the base case model and the test case model. For each run of the simulation, the marketplace parameters were set as follows:

Table 4. Experimental parameters used for simulation marketplace

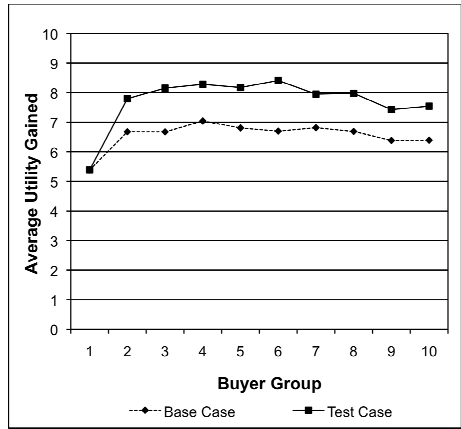
Parameter	Value	Parameter	Value
Buyers (initial)	50	New buyer group size	25
Sellers	100	Interval to add buyers (I)	100
Time steps	1000		
Bad Sellers (%)	15	Buyer Activity Constant (%)	5
Good Sellers (%)	15	Buyer Activity High (%)	10
Average Sellers (%)	60	Buyer Activity Medium (%)	70
Erratic Sellers (%)	10	Buyer Activity Low (%)	10
		Buyer Activity Very Low (%)	5
Honest Advisors (%)	70	Trust Threshold	0.75
Dishonest_high Advisors (%)	10	Advisor List Size	10
Dishonest_low Advisors (%)	10	Advisor Refresh Cycle (J)	100
Dishonest_erratic Advisors (%)	10		

By analyzing the average utility gained over the entire simulation run for each group of buyers (where group 1 is the initial buyer set and groups 2 through 10 are the sets of added buyers), we see that the test case model produces a higher average utility for all groups (Figure 1(a)). Furthermore, if we consider only the first ten time steps of the simulation, we see that the buying agents in groups 2 through 10 gain significantly more utility in the test case model than in the base case model (Figure 1(b)). This indicates that agents that are new to the system (novice agents) are immediately more effective when following the test case model over the base case model.

In the first ten time steps, however, group one performs almost identically in both models. This is due to the fact that all agents in the marketplace at that time are novice, and therefore there aren't any senior agents from whom to solicit advice. As a result, the agents in both models behave in the same manner for the first few time steps.

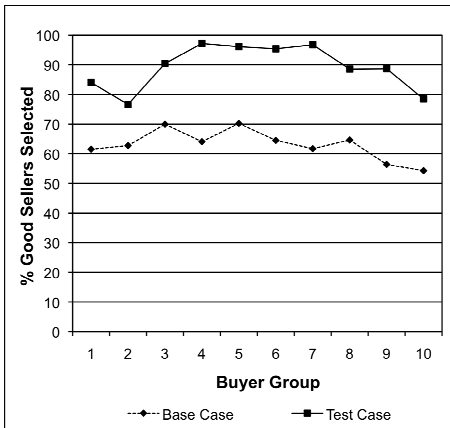


(a)

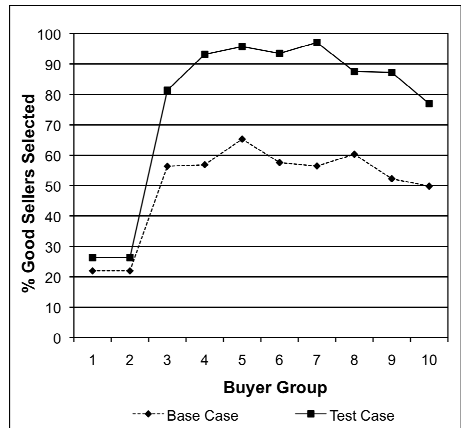


(b)

Fig. 1. Average utility gained by each buying group: (a) for all time steps of the simulation; (b) for the first 10 time steps that the buying agent group exists in the marketplace



(a)



(b)

Fig. 2. Average percent of sellers selected by each buying group that were good sellers after: (a) all time steps of the simulation; (b) first ten time steps for which the buyer group exists in the marketplace

By examining the choice of sellers by the buying agents, we see that the agents in the test case model were consistently significantly better at choosing good sellers than the buying agents in the base case model (Figure 2(a)). As with the average utility gained, by examining the behavior in the first ten time steps of each group's existence, we see that buying agents are able to make considerably better choices in their first few purchases by using the test case model over the base case model (Figure 2(b)). Again, the exceptions are with the first groups which yield similar results in the

first ten time steps regardless of the model used since the system does not at that time contain any experienced agents to use as advisors.

5 Discussion

The experimental results that we have presented show that the proposed parameters, agent lifetime and transaction count, can be used as part of a reputation model to improve results obtained in an agent's first few transactions, as well as over the lifetime of the agent. Overall, these parameters employed in the proposed manner led to better selection of seller agents, which in turn yielded more successful outcomes than in the base (random) case.

However, closer examination of the results reveals that while in most simulation runs, the test case model produced nearly perfect results for the selection of good sellers, in a small number of simulation runs, the results were poor. The latter runs were characterized by having an unusually high percentage of advisors who were both dishonest and had high seniority (i.e., had a long lifetime and a high transaction count) highly active. Therefore, the experimental results show that the test case model is vulnerable to scenarios where agents who have been in the system for a relatively long time conspire together to bloat their transaction counts (e.g., by engaging in a large number of transactions amongst themselves) then provide dishonest reputation ratings. The impact of this vulnerability can be reduced by adopting some of the techniques presented by others, such as maintaining a direct trust value for each advisor agent using a technique such as one of those described in [1, 3, 7].

We have shown that agent lifetime and transaction count can be used to improve reputation modeling, but that their use alone is not sufficient to cover all scenarios.

6 Conclusions and Future Work

In this paper, we have presented two new parameters that can be used to improve reputation modeling systems: agent lifetime and transaction count. Furthermore, through experimental results we have demonstrated that the use of these parameters in a simple reputation modeling system can (i) enable novice agents to construct an effective initial advisor list, thus attaining better results sooner, and (ii) enable agents to make improved trust decisions overall. We believe that the results presented here indicate that these parameters can be used to develop improved reputation models based on approaches presented in other research or entirely new approaches.

In future work, we intend to introduce these parameters into a more complex reputation modeling system to verify that they can be used to improve other models and to overcome some of the challenges revealed by the results analysis. This more elaborate model will be validated through testing in simulation against models presented by other researchers.

In addition, we intend to investigate how agent lifetime and transaction count can be used in computing the direct trust that one agent has in another. This approach would have two facets: first, the agent lifetime and transaction count of the agent being rated can be factored in the computation of the trust rating; secondly, we could introduce lifetime and transaction count as attributes that describe the relationship between the agent being rated and the agent performing the rating. In this second

scenario, the transaction count for the relationship would be similar to the values used in the experience-based approaches described in [7, 8, 10, 11, 12]; however, the introduction of the relationship lifetime would be entirely new.

Finally, we would like to investigate how agent lifetime and transaction count can be utilized to detect and avoid malicious behavior. For example, given the agent lifetime and transaction count for any agent, it should be possible to detect agents who leave the system to escape a bad reputation, then reenter and engage in a large number of transactions in a short period of time to falsely inflate their reputation.

References

1. Abdul-Rahman, A., Hailes, S.: Supporting Trust in Virtual Communities. In: Proceedings of the 33rd Hawaii international Conference on System Sciences. HICSS, January 04 - 07, vol. 6. IEEE Computer Society, Washington (2000)
2. Apple iPhone App. Store, <http://www.apple.com/iphone/appstore/>
3. Cohen, R., Regan, K., Tran, T.: Sharing Models of Sellers amongst Buying Agents in Electronic Marketplaces. In: Proceedings of the 10th International Conference on User Modeling—Workshop on Decentralized, Agent Based and Social Approaches to User Modeling (2005)
4. Dasgupta, P.: Trust as a Commodity. In: Gambetta, D. (ed.) *Trust: Making and Breaking Cooperative Relations*, electronic edition, Department of Sociology, University of Oxford, ch. 4, pp. 49–72 (2000)
5. Ebay, <http://www.ebay.com/>
6. Gambetta, D.: Can We Trust Trust? In: Gambetta, D. (ed.) *Trust: Making and Breaking Cooperative Relations*, electronic edition, Department of Sociology, University of Oxford, ch. 13, pp. 213–237 (2000)
7. Hang, C., Wang, Y., Singh, M.P.: An adaptive probabilistic trust model and its evaluation. In: Proceedings of the 7th international Joint Conference on Autonomous Agents and Multiagent Systems. International Conference on Autonomous Agents, Estoril, Portugal, May 12 - 16, vol. 3, pp. 1485–1488. International Foundation for Autonomous Agents and Multiagent Systems, Richland (2008)
8. Huynh, T.D., Jennings, N.R., Shadbolt, N.: Developing an integrated trust and reputation model for open multi-agent systems. In: Proceedings of the 7th International Workshop on Trust in Agent Societies, New York, USA, pp. 65–74 (2004)
9. Ramchurn, S.D., Huynh, D., Jennings, N.R.: Trust in multi-agent systems. *Knowl. Eng. Rev.* 19(1), 1–25 (2004)
10. Reece, S., Rogers, A., Roberts, S., Jennings, N.R.: Rumors and Reputation: Evaluating Multi-Dimensional Trust within a Decentralized Reputation System. In: Proceedings of the Sixth Intl. Joint Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2007), pp. 1063–1070 (2007)
11. Ries, S.: Certain trust: a trust model for users and agents. In: Proceedings of the 2007 ACM Symposium on Applied Computing. SAC 2007, Seoul, Korea, pp. 1599–1604. ACM, New York (2007)
12. Sabater, J., Sierra, C.: REGRET: Reputation in gregarious societies. In: Proceedings of the Fifth International Conference on Autonomous Agents, Montreal, Canada, pp. 194–195. ACM Press, New York (2001)
13. Yu, B., Sycara, K., Singh, M.: Developing Trust in Large-Scale Peer-to-Peer Systems. In: Proceedings of First IEEE Symposium on Multi-Agent Security and Survivability, MASS 2004 (2004)

Towards a Methodology for Representing and Classifying Business Processes

Hafedh Mili, Abderrahmane Leshob, Eric Lefebvre, Ghislain Lévesque,
and Ghizlane El-Boussaidi

LATECE Laboratory, Université du Québec à Montréal,
B.P 8888, succursale Centre-Ville, Montréal (Québec) H3C 3P8, Canada
first.last@uqam.ca, eafrolefebvre@videotron.ca

Abstract. Organizations build information systems to support their business processes. Some of these business processes are industry or organization-specific, but most are common to many industries and are used, modulo a few modifications, in different contexts. A precise modeling of such processes would seem to be a necessary prerequisite for building information systems that are aligned with the business objectives of the organization and that fulfill the functional requirements of its users. Yet, there are few tools, conceptual or otherwise, that enable organizations to model their business processes precisely and efficiently, and fewer tools still, to map such process models to the software components that are needed to support them. Our work deals with the problem of building tools to model business processes precisely, and to help map such models to software models. In this paper, we describe a representation and classification of business processes that supports the specification of organization-specific processes by, 1) navigating a repository of generic business processes, and 2) automatically generating new process variants to accommodate the specifics of the organization. We present the principles underlying our approach, and describe the state of an ongoing implementation.

1 Introduction

Organizations build information systems to support their *business processes*. One is tempted to infer that two organizations that employ the same business processes should be able to use (or reuse) the same IT infrastructure. At a very high-level of abstraction, this may very well be the case (e.g. BIAIT [1]). However, reuse at the IT strategic or enterprise architecture level does not translate into reuse at the more concrete software artifact level (models, at all levels, and code), where most of the development and maintenance resources are spent. There are many reasons for this. First, there is a wide variety of business processes for doing anything, from purchasing, to inventory management, to billing, or hiring, which may coincide on the fundamentals, but differ in the detail. Further, for any given business process, there are different levels of IT support, ranging from a simple recording of the activities of an essentially human business process, to performing the clerical steps of the process, and recording the others, to a full process automation. The BIAIT methodology, for example, recognizes some of these variations and encodes them in half a dozen binary decisions.

However, this comes far short of capturing all of the possible variations in the business processes, and in the level of support provided by the IT system.

Add to the above variations the great variety of target domains, or industries, such as banking, insurance, manufacturing, pharmaceuticals, etc. But how different are the business processes across domains, really? The process of *selling* computers is similar to the process of *selling* cars, much more so than to the process of *manufacturing* computers. In fact, because cars and computers share many characteristics—both are tangible, somewhat high-priced, manufactured, configurable products—the processes are nearly identical. What distinguishes the two is the domain vocabulary (computers vs. cars, processors vs. engines, etc.). A good fraction, if not the majority, of the business processes of an organization do not depend on the industry within which it operates. However, the *analysis models* of the information systems that support them will be domain-specific.

Our long term objective is to characterize the transformation from business process to software with enough precision to be able to instrument it with tools that help business and system analysts, working together, to generate a first sketch of the target software system based on a precise model of the business processes that it supports. Traditional approaches to this problem are catalogue-based: we build a catalog of software components that are somehow *indexed* by the *elementary* business processes that they support. To differing degrees, this is the approach taken by the IBM San Francisco initiative and by SAP through its ‘blueprint’. The San Francisco initiative suffers from the relatively low granularity of the components, creating a significant semantic gap between the business process level, and the software component level. Further, lots of glue is needed to assemble the low-granularity components, diminishing the reuse effectiveness of the approach. For the case of SAP, the business processes are at the right level of granularity, but the mapping from business process to software is embodied in proprietary tools, and there is still quite a bit of customization required.

Our approach to this problem consists of precisely characterizing and codifying the three sources of variability that we mentioned:

1. Process variability: accommodating differences in business processes to accomplish the same business objective.
2. Domain variability: accommodating differences between application domains.
3. Automation variability: accounting for the fact that different information systems will offer different levels of automation for the same processes.

Our approach to handling the first two sources of variability relies on a combination of, a) a catalog of generic business processes, b) a representation system for such business processes that supports a number of specialization operators enabling us to generate on-the-fly specializations, and c) a mapping procedure that enables us to *instantiate* a generic business process for a particular domain. The difference between our approach and other catalogue-based approaches (e.g. [8]) is that our catalogue does not need to be exhaustive: we can generate new process variants (specializations) on-the-fly, as opposed to having to encode them manually.

In the next section, we describe the process modeling language that we will be using. Section 3 talks about how we handle process variability and domain variability. In particular, we discuss our approach to generating and representing process specializations.

Section 4 describes the design of a toolset for the representation and classification of business processes. Section 5 describes the current implementation. We conclude in section 6 by highlighting directions for future work.

2 Process Modeling

Curtis defined a process as a partially ordered set of *tasks* or *steps* undertaken towards a specific goal [3]. Hammer and Champy define *business processes* as a set of *activities* that, together, produce a result of value to the customer [5]. The workflow management coalition defines business processes as “a set of one or more linked procedures or activities which collectively realize a business objective or policy goal, normally within the context of an organizational structure defining functional roles and relationships.” [9],[10]. We adopt the definition embodied in the meta-model of Fig. 1. The **activities** of a business process are performed by **actors** playing particular **roles**, consuming some **resources** and producing others. Activities may be triggered by **events** and may, in turn, generate events of their own. The **activities** of a process may be linked through **resource dependencies** (producer-consumer dependencies) or **control dependencies** (one activity triggering another). The **actors** operate within the context of **organizational** boundaries. Organizations perform specific business **functions**. Roles can support **functions**. We will refer again to this meta-model when we present our classification procedure.

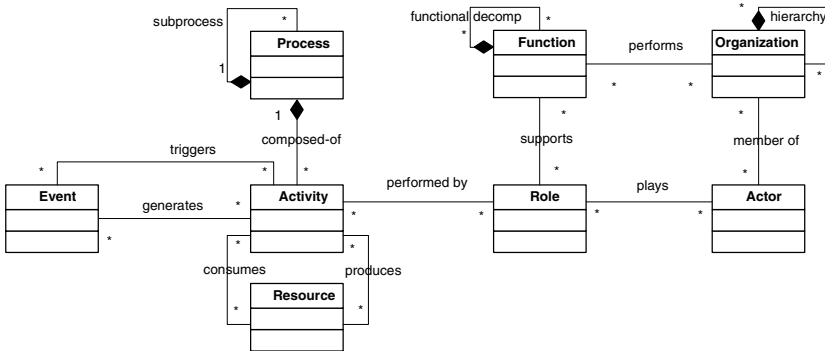


Fig. 1. A first-cut business process metamodel

As our process metamodel suggests, there are a number of things to represent about a process. Curtis argued that there are four distinct views [3]: (1) *The functional view* presents the functional dependencies between the process elements, such as producer-consumer dependencies. (2) *The dynamic view* provides sequencing and control information about the process, i.e. when certain activities are performed, and how. (3) *The informational view* includes the description of the entities that are produced, consumed or otherwise manipulated by the process. These entities include pure data, artifacts, products, etc. (4) *The organizational view* describes *who* performs each task or function, and *where* in the organization (functionally and physically).

Most object-oriented modeling notations cover the first three views. What is new, from an ontological point of view, is the *organizational* view, which includes a description of the *participants* in the process as well as a description of the physical (location) and organizational context within which this process is conducted. Further, whereas the informational view in object-oriented modeling represents only data entities, the informational view of business process modeling may represent tangible resources and artifacts that are used and produced by processes.

We studied a dozen or so process modeling languages that originated from a variety of scientific traditions (see e.g. [11]). Because we want to *ultimately* map our process models to information system object models, we used UML 2 as a basis, and used its extension mechanisms to introduce the organizational view.

3 An Approach to Business Process Classification

3.1 Principles

For the purposes of illustration, we will use the example of an ordering process. Ordering starts by first filling out a request for a product which then goes through a budgetary approval process. If it is approved, it goes to purchasing, who will identify suppliers for the product. Once a supplier is found, then a purchase order to *that supplier* is created and sent. When the product is received, it is sent to the requester. Then payment is made. Fig. 2 shows a simplified functional view of the process.

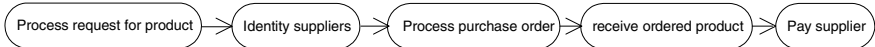


Fig. 2. The functional view of a basic ordering process

This process is independent from the application domain, and at this level of abstraction, it can be used to order pencils or computers or airplanes. There are many variations of this process, notwithstanding the target application domain. For example, for on-line purchases, we typically pay before “receiving” the product, because of the anonymity (and impunity) of the internet. Second, if the buyer has a running contract with a supplier, then they do not look for suppliers each time: they order directly from that supplier. Third if the requester is also the decision maker, they do not need to ask for approval: they can just go ahead and order the product.

These are all variations that do not depend on the target industry or application domain. Naturally, we expect the software applications that support the purchasing process to exhibit similar variations. This raises two questions:

1. Is there a way to organize existing business processes in a specialization hierarchy that makes it easy for users to navigate to find the business process that best fits their organization,
2. Is there a way to *generate* some of these specializations on-the-fly based on some catalog of elementary specializations,

We discuss each question briefly. Subsection 3.2 presents our approach.

There have been many initiatives aimed at cataloguing generic business processes, each proposing classifications of their own, including the MIT process handbook, the IBM San Francisco project, and various analysis pattern catalogues. These classifications are for the most part high-level, and refer to broad functional areas such as *production*, *logistics*, *support*, or *planning* (e.g. [2],[7]). These classifications are also *descriptive* in the sense that they are based on external properties of the process (*meta-data*) as opposed to *structural* classifications, which are inherent in the structure of the models—and can be computed from it. The MIT process handbook uses a descriptive classification, in addition to a question-based classification discussed in the next section. Descriptive classifications require little automation, and are easy to implement. They are, however, labor intensive. Structural classifications would help us answer the second question, i.e. generate on the fly process specializations based on a catalogue of generic processes and a catalogue of elementary specializations.

Fig. 3 shows a simplified model from the *informational view* of the ordering process. As mentioned earlier, the ordering process would depend on the existence of a *contract* between the buyer and the appropriate supplier regarding the terms of purchase (price, delivery delays, defect return policy, etc.), which will obviate the need for searching for a supplier. Second, the reception of the product will depend on whether the product is a tangible product (a chair, a computer, a book), or a non-tangible product, or *service*, such as internet access, phone service, etc.

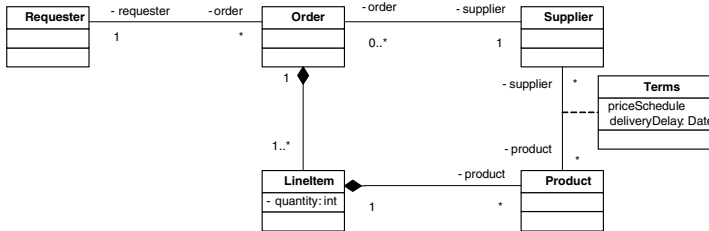


Fig. 3. Basic (partial) informational view for the ordering process

Figure 4 shows a new object model (informational view) that accommodates both of these changes. The new model is similar to the original one, with two differences (noted in grey boxes): 1) we added a class (**Contract**) and two associations between the new class and existing ones, and 2) we specialized an existing class (**Product**) into two subclasses (**TangibleProduct** and **Service**). This simple example raises a number of points that we discuss below.

Let us first start with the specialization of **Product** into **TangibleProduct** and **Service**. We, in the object world, are familiar with these kinds of specializations. In framework speak, these are called hotspots, which are well-defined points of extension in object models using well-defined extension mechanisms; in this case, sub-classing. Sub-classing or sub-typing only covers the simplest cases. With the contract example, we are adding a class and two associations, and there is no way of guessing that the addition actually *specializes* the process. Third, adding a contract between buyer and supplier actually removes one step from the functional view. It also modifies the dynamic view accordingly.

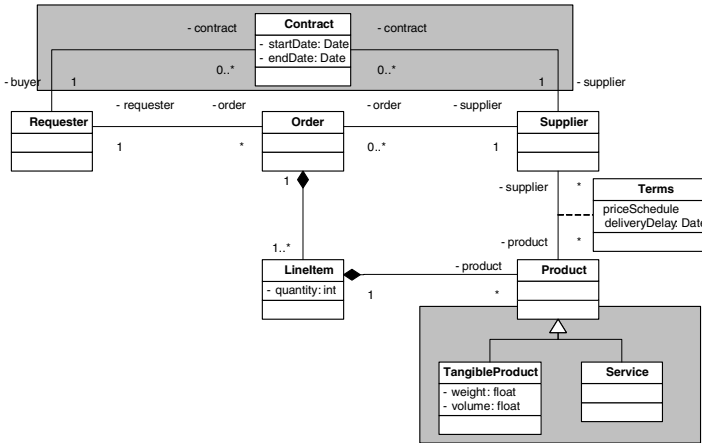


Fig. 4. The informational view for a specialization of the purchasing process

This illustrates a problem first raised by Lubars in 1992 [6]. Objects have been originally sold on the intuitive belief that “small” changes in requirements result in “small” changes in the corresponding program, thanks to inheritance and encapsulation. Lubars showed that this *may* be true for the object model, but not so for the dynamic model, for example: small changes in requirements can result in dramatic changes in the dynamic model [6]. What all this means is the following:

1. what we might intuitively refer to as *process specialization* may have a simple expression in one of the four views, but not necessarily in the others
2. the specialization operators depend on the view, and may not be related to the object-oriented specialization or extension operators
3. one specialization will affect several views differently.

The answer to our second question is then, yes, it might be possible to generate process specializations on the fly using a catalogue of elementary specialization operators, but that catalogue will have to include far more than the typical object-oriented ones (composition, inheritance).

3.2 Classification Using Metamodel Hotspots

Our analysis of the business process classification problem above showed that there are no simple specialization operators that can be applied on any of the views that would yield systematically meaningful specializations. Some previous work has used *questions* to derive specializations of processes. Carlson argued that the purpose of any organization is to offer a product or a service to a client, and hence, an information system that supports the organization would need to manage this “ordering” process [1]. The data and the operations supported by the information systems depend on the business model and on the way the organization works.

Carlson had reduced these variations to the answers to seven questions whose answers (yes/no) determine the kind of process, and thus the information system needed

to support it. We show below a couple of questions, and illustrate their implications on the business process and on the corresponding information system:

- *Does the supplier send an invoice to the customer, or does the customer pay for the product/service cash (or equivalent)?* If the supplier sends an invoice, we have an invoicing process and a payment process with checks, wire transfers and the like. Also, the information system will need to keep information about the customer, their billing address, and their banking information. If the customer pays cash, no records need to be kept of the customer.
- *Are the prices negotiated, i.e. they differ from one customer to another, or are they fixed?* Negotiated prices mean contracts, price schedules per customer, etc.
- *Is the product or service leased to the customer by the supplier, who conserves all property rights, or is property transferred to the customer?* If the product is leased, the organization needs to keep track of the leasee and manage the product or service throughout its lifecycle. This also has major implications on accounting.

Lefebvre used a variation of these questions to help identify *software* component archetypes [2]. Notwithstanding the fact that BIAIT's seven questions may not be orthogonal—they are not—the questions are fairly coarse-grained, and alone cannot capture the level of detail required for the processes to be able to generate the corresponding information system models. The MIT process handbook also used questions to specialize processes [8]. However, the questions are process-specific. Using process-specific questions has the advantage that both the questions and the resulting specialized processes are precise. It has the disadvantage that the classification is ad-hoc and cannot be generalized: whoever specifies a generic business process has to classify and encode all the variations that would make sense, manually. Further, we cannot generate process specializations on the fly.

By going over a number of processes from the MIT process handbook, and the associated questions, we realized that the questions are about the *roles* involved in the process (customer, supplier), the nature of the *resources* produced and consumed by the various *activities* (product, service, tangible product), or the *organization* within which activities are taking place. Thus, we can frame our questions generically in terms of entities and associations in the process metamodel, and then *instantiate* them for specific processes—instances of the process metamodel—to get process-specific questions. Some of these questions are more related to the informational view, while others are related to the organizational view, while yet others are more related to the functional and dynamic view. We reproduce in Fig. 5 the partial business process metamodel where we outlined the model fragments included in each view.

We show below a couple of generic questions, how they impact a process, and see how they are instantiated for a specific business process. The view is shown between parentheses:

- *Can an actor play several roles within the process (organizational)?* when an actor plays several roles within the same process instance, the underlying process is generally simplified. In our purchasing example, we have three roles within the purchasing organization involved in the creation of the purchase order: the requester (end-user), the person responsible for the budget, and the purchasing agent. If the requester and the budget person are the same, we don't need approval.

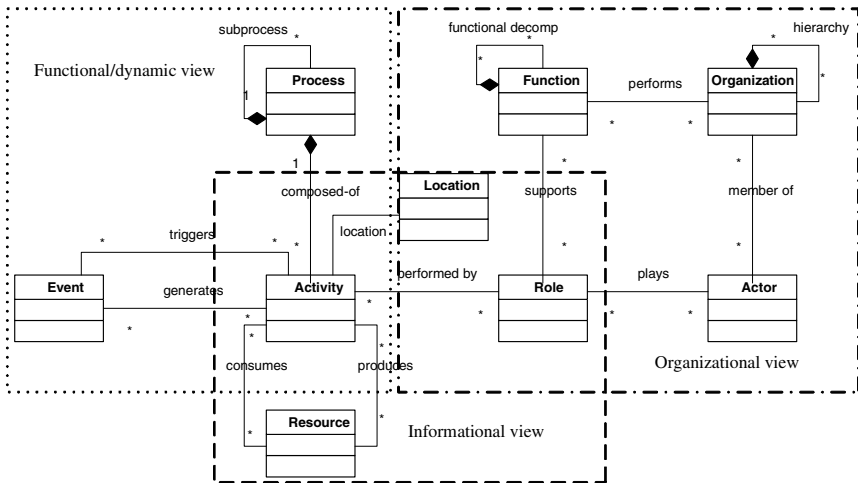


Fig. 5. A partitioning of the business process metamodel

- *Is information about the actors recorded (informational)?* when information about an actor is recorded, the actor is represented internally in the system. Also, activities will be logged in the system. In the purchasing example, this question will be instantiated into several questions, one per actor: a) is information about the requester recorded?, b) is information about the budget approver recorded?, c) is information about the purchasing agent recorded? A yes in each case will have implications on the data and the processing of the application.
- *Does the process execution follow an agreement of some sort (functional)?* if there is an agreement, it means either that process parameters can be obtained from that agreement, or that process execution needs to be validated against the agreement. In the purchasing example, we have the case where a binding agreement exists between the purchaser and a supplier, which simplifies the search for suppliers and initializes some parameters (e.g. price), and the case where no such agreement exists, in which case the order goes to the lowest bidder, for example.

We have identified fifteen (15) questions in all, five organizational, four functional, and six informational. Some of the informational questions have to do with the nature of the resources (tangible vs. non-tangible, perishable or not, degradable through consumption or not, limited quantity or not).

Once we have identified the questions, we have to determine the effect of the answer on the corresponding process models, and more specifically, on each view. Naturally, the questions may impact some views more than others. For each question, we need to develop a set of transformations per view. Some of these transformations consist of removing model fragments that follow a specific pattern, as in removing coordination activities between roles played by the same actor. Others consist of adding model elements (entities, associations, processes) to model fragments that satisfy a specific pattern, in much the same way that we apply analysis or design patterns to existing models. In fact, we are using some of the published analysis and process patterns to this end [2], [4].

4 Designing a Tool for Process Modeling and Specialization

We have started developing a tool set to experiment with the ideas presented in section 3.2. In this section, we present the design principles behind our toolset. In subsection 4.1, we present the concept of operations of our toolset, i.e. how we envision it to be used by process modellers. Subsection 4.2 looks at the representation of process models, specialization questions, and process specializations.

4.1 Concept of Operations

At the core of our toolset is a repository of business processes that process analysts and modelers can browse to find the process that most closely matches the needs of their organizations. In those cases where the best match that is found in the repository is not close enough, process analysts are expected to specialize one of the existing processes of the repository to obtain a process that more closely matches the needs of their organization. Initially, we will have to populate the repository with a set of generic processes that we could take from the MIT process handbook or similar public domain sources. It is expected that usage of our toolset will continually enrich the repository. Eventually, we expect to get to the point where process specialization becomes a rare occurrence, as process modelers will likely find what they need in the repository. Figures 6-a and 6-b illustrate this process.

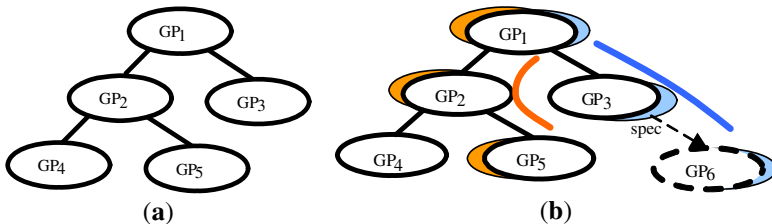


Fig. 6. (a) We start with a catalogue of generic business processes, GP₁ through GP₅. (b) Modellers find the process they need by using one of the stored ones as-is (e.g. GP₅), or by specializing an existing process (GP₃) using specialization operators to obtain a new process that will be stored in the repository (GP₆).

We envision the proposed system as a pair of applications that access the process repository: a) a web application that enables process modelers to consult the repository in a read-only mode, and b) a desktop application for specializing—and more generally, editing—processes. The main entry point of our system is the web application, which enables process modelers to navigate the process repository (forest). When needed, a process modeler can select a process to specialize, and that takes them to the desktop application. Fig. 7 illustrates this. The desktop application, shown in the left hand side, is identified as an “Eclipse EMF application”. Indeed, as explained in section 5, we will use the Eclipse Modeling Framework (EMF) to implement our process metamodel, and use EMF based tools, (graphical) editors, and code generators to build and specialize processes built using that metamodel.

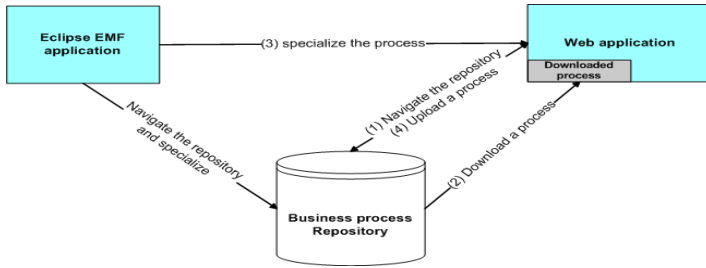


Fig. 7. Proposed specialization system

4.2 Representing Process Models, Questions, and Specializations

Process Models and Process Views. A tool that supports the representation and classification of business processes needs to support each one of Curtis’s four views (i.e. *informational, functional, dynamic, and organizational*) Each view consists of a set of model elements appropriate for that particular view. For example, the informational view will contain entities (classes) and associations, whereas the dynamic view will contain activities, events, and transitions. And so forth. As mentioned above, the answer to a question will specialize each one of the views using view-specific specialization operators. A simple metamodel is shown in Fig. 8. We will complete the various pieces in the subsequent paragraphs and in section 5.

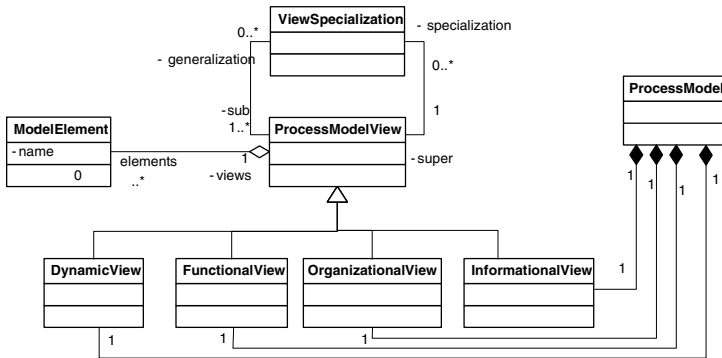


Fig. 8. Business process views

Question model. We can think of generic questions as functions that take a number of parameters and that have a return value. The parameters have *types* which are process metamodel entities, and that correspond to what we called *metamodel hotspots*. The return value is the answer to the question. Typically, questions will have Boolean answers (true/false), but in the general case, we will say that the answer to a question belongs to an “answer type”.

Let us start with the example of the generic question “*does the process execution follow an agreement or a contract between Actor and Actor*”. Symbolically, we can think of this question as the function:

boolean processExecutionFollowsContractBetween(**Actor** actor_1, **Actor** actor_2);
Here, the type **Actor** refers to the class that is part of the business process metamodel (see Figures 1 and 5).

This function can be instantiated for a specific business process to yield process-specific questions. Consider our ordering process of section 3. That process—its organizational view—includes two actors: a **Requester** and a **Supplier**. The process-specific questions consist of all of the possible invocations of the function above by binding the formal parameters “actor_1” and “actor_2” to actual actors from my ordering process. Because our process has two actors, we get four possible process-specific questions, corresponding to four possible “function invocations”:

- 1) processExecutionFollowsContractBetween(**Requester**, **Requester**)
- 2) processExecutionFollowsContractBetween(**Requester**, **Supplier**)
- 3) processExecutionFollowsContractBetween(**Supplier**, **Requester**)
- 4) processExecutionFollowsContractBetween(**Supplier**, **Supplier**)

The corresponding natural language transcriptions of these functions are:

- 1) *does the process execution follow an agreement or a contract between **Requester** and **Requester***
- 2) *does the process execution follow an agreement or a contract between **Requester** and **Supplier***
- 3) *does the process execution follow an agreement or a contract between **Supplier** and **Requester***
- 4) *does the process execution follow an agreement or a contract between **Supplier** and **Supplier***

If we apply the generic question to a process model that involves three actors, this will result into *nine* different process-specific questions (3 X 3), and so forth.

Clearly, some of these instantiations are not very interesting, and it would have been easy to identify them. For example, in this case, it does not make sense to instantiate the question for a pair of identical actors (<**Requester,Requester**> and <**Supplier,Supplier**>). Further, this question is “symmetrical” in the sense that $f(x,y) = f(y,x)$ for a given x and y . Given this information, when we apply the generic question to the ordering process, we should get a single interesting process-specific question, “processExecutionFollowsContractBetween(**Requester**, **Supplier**)”, or, in English, “*does the process execution follow an agreement or a contract between **Requester** and **Supplier***”. Accordingly, our representation of generic questions includes a *filter* that applies to parameter bindings to eliminate unacceptable or redundant bindings.

Fig. 9 shows our model for representing generic questions and process-specific questions. The **GenericQuestion** class has the attributes name, description, core, and filter. The ‘core’ attribute is used to represent the natural language template of the question. In this case, it is the string “does the process execution follow an agreement or a contract between {0} and {1}”, where {0} and {1} refer to the positional parameters 0 and 1. The ‘filter’ attribute represents the instantiation filter as illustrated in the above example. The parameters of a **GenericQuestion** are represented by the association class **Parameter** which indicates the parameter name, position (0, 1, etc.). The type of the parameter is represented by the end class **ModelElementType**, which stands for process metamodel entities. The lower part of the model shows how process

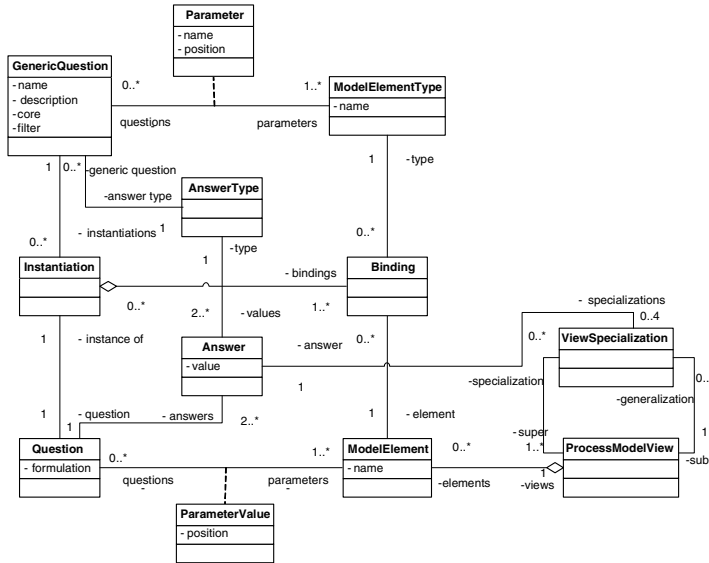


Fig. 9. Question model

specific questions (**Question**) are instantiated from **GenericQuestion** by binding (class **Binding**) formal parameters to actual process model elements. A given **Question** can have two (yes/no) or more answers (**Answer**), and each answer can potentially result into a view specialization¹.

The specialization model. As we have shown in Fig. 9, a specific answer (e.g. “yes” or “no”) to a process-specific question, can yield a specialization of one or more of the process model views. We think of these specializations as the application of *view-type specific, generic transformations to process model views*. These transformations are *view-type specific* because each view type (informational, functional, dynamic, and organizational) has its own transformations. They are *generic* because they are meant to apply to *all* process models—or process model views. Symbolically, with our ordering process and our question about contracts, we can express the specialization (transformation) of the informational view this way:

If (processExecutionFollowsContractBetween(actor_1, actor_2) == true) **Then**

- 1) Create a new class cls with name ‘**Contract**’ with attributes ‘terms’, ‘startDate’, ‘endDate’, and add it to the informational view;
- 2) Add an association between cls and actor_1 labeled “binds” with cardinality to the informational view;
- 3) Add an association between cls and actor_2 labeled “binds” with cardinality to the informational view;

¹ Not all answers result into a specialization. For example, if the answer to our question “*does the process execution follow an agreement or a contract between Requester and Supplier*” is “no”, we keep the same process. Only when the answer is “yes” do we get a more specialized process. The 4 cardinality in the 0.4 is due to the fact that an answer can specialize a process along 0, 1, 2, 3, or 4 views.

We represent the transformations in terms of a set of **TransformationRule**'s. Given a (i.e. one) generic question, an (1) answer, and a view type (1, e.g. informational), we can have *several* transformation rules. In our example here, we expressed the transformation with a single rule, but in the general case, we can have several. Fig. 10 illustrates this. It is actually **the codification of these generic transformations that represents the major conceptual difficulty in our approach**. We think of the **ViewSpecialization** entity as a *trace* of the actual transformation rules as they were applied to a particular view. The “trace” relationship is represented by the many-to-many association between **TransformationRule** and **ViewSpecialization**.

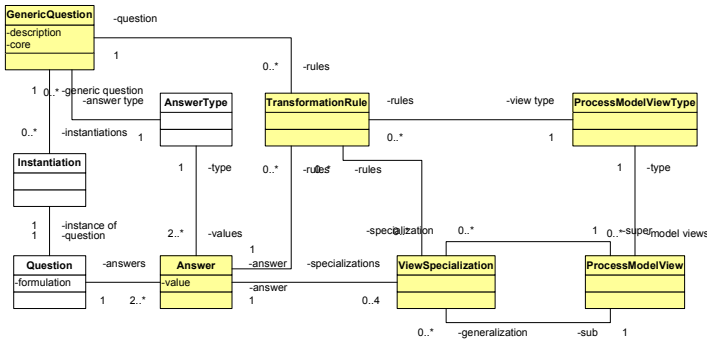


Fig. 10. Process specialization model

5 Implementing Process Modeling and Specialization

In this section, we describe the state of the current implementation of our toolset for modeling and specializing processes. Referring to the concept of operations shown in Fig. 7, so far we worked only on the desktop modeling application; we have not started work on the web application. As hinted at in Fig. 7, the desktop application is based on the Eclipse platform, and uses the Eclipse Modeling Framework (EMF) as the foundation for implementing the process metamodel and the modeling tools.

5.1 Implement the Business Process Metamodel

Figures 1 and 7 showed a (much) simplified business process metamodel. For our modeling tool, we decided to implement a more realistic, preferably standardized, business process metamodel. However, there were many standards to choose from (see [11]). The OMG set-out to define ... a standard for unifying these standards, and it came up with the Business Process Definition Metamodel (BPDM, [12]). BPDM is meant to capture the common constructs of the various process modeling languages, while supporting the representation of notation-specific constructs. Consequently, BPDM has a fine conceptual granularity, and we felt that it was far too granular for our purposes. Accordingly, we decided to implement a subset of BPDM.

For the purposes of this paper, we show a *simplified* version of that *subset* in Fig. 11. A Process is defined as a set of **Step**'s sequenced using **ProcessingSuccession**'s.

Steps can correspond to actual work (**WorkDefinitionStep**), e.g. “Fill out purchase order”, or control steps (e.g. a fork or a join) or an event step (e.g., a process start or end). Work definition steps consume and produce resources (**ResourceProduct**). **Activity**'ies (elementary work definition steps) are performed by **Actor**'s playing certain roles (**RolePerformer**).

We implemented our business process meta-model using the *Eclipse Modeling Framework*TM (EMF). EMF is a Java modeling framework that implements a core subset of the Meta Object Facility (MOF) API in a package called *ecore*. EMF provides functionality to serialize models implemented with the *ecore* package into XMI (XML Metadata Interchange) files. So far, we have implemented only the informational and dynamic view.

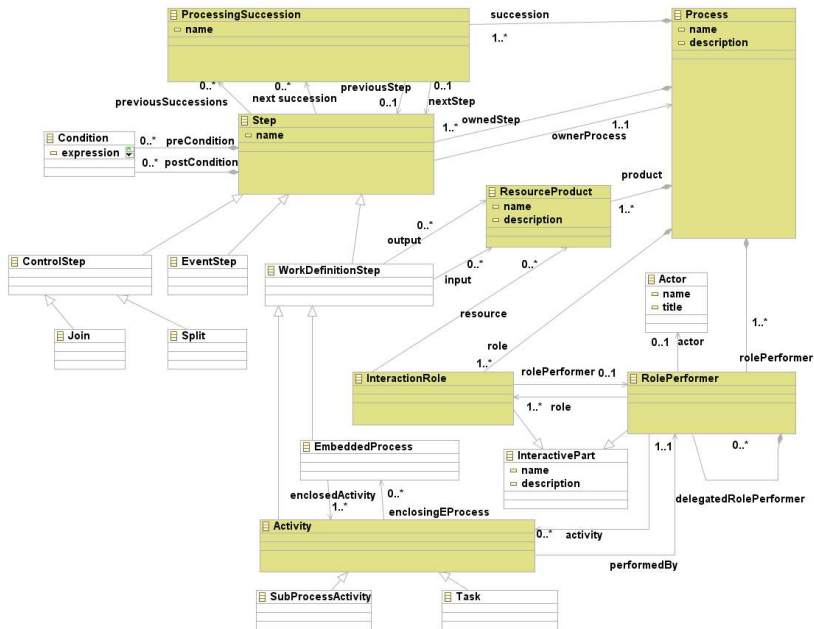


Fig. 11. A simplified view of the implemented business process metamodel

5.2 The Question Model

The generic question metamodel (Fig. 9) was implemented using an XML schema. Hence, generic questions are represented using an XML document. Fig. 12 shows the document that represents the question “*does the process execution follow an agreement between RolePerformer and RolePerformer*”. The string that represents the question template is represented by the <core> tag. Note the fillers “{0}” and “{1}”, which are like macro variables that stand for the first and second parameters. When this generic question is instantiated for a specific pair of **RolePerformer**'s, the names of these role performers will replace “{0}” and “{1}” in the string. Note also that the instantiation filter is represented by an attribute of the <arguments> tag. The representation of process-specific questions follows a similar structure.

```

<question>
  <name>Binding agreement between role performers</name>
  <description>The process follows an agreement between role performers
  </description>
  <core answerType="java.lang.Boolean" >
    Does the process execution follow an agreement between {0} and {1}?
  </core>
  <arguments filter = "!reflexive, symmetric">
    <argument>RolePerformer</argument >
    <argument>RolePerformer</argument >
  </arguments >
</question>

```

Fig. 12. The representation of generic questions

5.3 Handling Specialization

Recall that process specialization is performed by applying transformations to existing model views, based on the answers to specific questions. We showed in the metamodels of Figures 9 and 10 that each `<question,answer>` pair may specialize from zero (i.e. no specialization) to four views. We also showed in Fig. 10 that the transformation rules are `<question, answer, view>` specific. When we started implementing model specialization, we quickly realized that, practically, not all process specializations are worth storing in the repository. Indeed, a process modeler might submit an existing process from the repository to a series of questions, and might be interested only in the final process resulting from applying a string of `<Questioni, Answeri>` specializations, for $i=1..n$. This means that intermediary process models, those resulting from `<Question1, Answer1>`, `<Question2, Answer2>`, ..., `<Questionn-1, Answern-1>` are not stored: only the starting point and the end point are stored. Accordingly, the specialization link between a process and its specializations is not *always* indexed by a single `<question,answer>`: it can be a *sequence* of `<question,answer>` pairs, to which we refer as *process-specific question paths*.

With regard to the transformation rules, we used JBOSS DROOLS, an open-source object-rule production system. Space limitations do not allow us to illustrate our rules. But roughly speaking, the condition part of a rule in our context is satisfied if the rule finds a question with answer "TRUE"² while the action part performs the corresponding specialization (e.g. for the *ordering with contract* example, the action part creates the **Contract** class and links it to the classes **Requester** and **Supplier**).

6 Discussion

Organizations build information systems to support their business processes. Some of these business processes are industry or organization-specific, but most are common to many industries and are used, modulo a few modifications, in different contexts. A precise modeling of such processes would seem to be a necessary prerequisite for building information systems that are aligned with the business objectives of the organization

² The answer type is String, instead of Boolean. String makes it easier to handle all simple answer types (boolean, int, enumerated values, etc.).

and that fulfill the functional requirements of its users. Yet, there are few tools, conceptual or otherwise, that enable organizations to model their business processes precisely and efficiently. In this paper, we proposed a representation and classification of business processes that supports the specification of organization-specific processes by, 1) navigating a repository of generic business processes, and 2) automatically generating new process variants to accommodate the specifics of the organization.

Our process for specializing process models is question-driven. There are innumerable ways of specializing business processes, but like others before us (e.g. [1], [2], [8]), we argued that process variability can be captured in a relatively small number of questions. Unlike the MIT process handbook approach [8], our questions are generic. Unlike the BIAIT approach [1], our questions are precise enough that they can be linked to specific model entities. Finally, unlike Coad et al.'s archetypes [2], our questions handle all process model views. Central to our approach is the assumption that generic specialization operators can be defined that can specialize any process based on the answer to one of the generic questions. We have had some empirical evidence that this might be true, at least for some questions and class of models.

In this paper, we discussed the design principles of a tool set based on these ideas, and presented the current status of an implementation. This implementation showed the plausibility and *computational feasibility* of the approach. The conceptual work of identifying and codifying the transformation rules has just begun. We believe that this line of work will help in understanding the nature of variability and specialization in business processes and, ultimately, in information models.

References

1. Carlson, W.M.: Business Information Analysis and Integration Technique (BIAIT)-the new horizon. *Data Base* 10(4), 3–9 (1979)
2. Coad, P., Lefebvre, E., Luca, J.: *Java Modeling In Color With UML*. Prentice-Hall, Englewood Cliffs (1999)
3. Curtis, B., Kellner, M.I., Over, J.: Process modeling. *Com. of ACM* 35(9), 75–90 (1992)
4. Fowler, M.: *Analysis Patterns*. Addison-Wesley, Reading (1997)
5. Hammer, M., Champy, J.: *Reengineering the Corporation*. Harper Business (1993)
6. Lubars, M., et al.: Object-Oriented Analysis for Evolving Systems. In: *ICSE*, pp. 173–185 (1992)
7. Maiden, N.A., Sutcliffe, A.G.: Exploiting Reusable Specifications Through Analogy. *Com. of the ACM* 35(4), 55–64 (1992)
8. Malone, T.W., et al.: Tools for inventing organizations. *Manag. Sc.* 45(3), 425–443 (1999)
9. Workflow Management Coalition. Doc. WfMC-TC-1016-P (1999)
10. Workflow Management Coalition. Doc. WfMC-TC-1011 (1999)
11. Mili, H., Tremblay, G., Bou Jaoude, G., Lefebvre, E., El Abed, L.: Business Process Modeling Languages: Sorting Through the Alphabet Soup. *ACM Computing Surveys* (2008)
12. Business Process Definition Metamodel. OMG Document bmi/2007-03-01 (2007)

Typing for Conflict Detection in Access Control Policies

Kamel Adi¹, Yacine Bouzida¹, Ikhlass Hattak¹, Luigi Logrippo¹,
and Serge Mankovskii²

¹ Security Research Laboratory
Computer Science and Engineering Department
Université du Québec en Outaouais, Québec, Canada

² CA Labs
125 Commerce Valley DR W
Thornhill ON, L3T 7W4, Canada

Abstract. In this paper we present an access control model that considers both abstract and concrete access control policies specifications. Permissions and prohibitions are expressed within this model with contextual conditions. This situation may lead to conflicts. We propose a type system that is applied to the different rules in order to check for inconsistencies. If a resource is well typed, it is guaranteed that access rules to the resource contain no conflicts.

1 Introduction

Most current access control models use authorizations to express the ability of a subject to perform an action on an object. In their basic form, authorizations are expressed with sets of triples, called rules, of the form $\langle \text{subject}, \text{action}, \text{object} \rangle$ meaning that a certain subject (user, process) is permitted to perform an action (an available operation) over an object (a resource in the target system). Such rules are grouped to form policies. Additional flexibility can be obtained by combining prohibitions (negative authorizations) with permissions. Also, in addition to concrete rules involving specific subjects, actions, and objects, it should be possible to specify abstract rules defined on classes of such entities. Flexibility can be further increased by making the application of rules conditional to predicates on contextual information (e.g. a rule is active at certain times only). The Security Officers (SOs) can then express general positive contextual authorizations and then add prohibitions to express exceptions. For instance, nurses can be authorized to consult medical records except those corresponding to emergency situations. We note that some access control models, such as RBAC [16], can express only positive authorizations while others, such as OrBAC [1], can express both positive and negative authorizations.

Complex sets of security policies can contain conflicts, since such sets can consist of thousands of rules that SOs can change over time. Conflicts can result in situations where a rule allows access, while another rule denies it. Sets

of policies can contain conflict resolution strategies, on which there is considerable research [4,5,8,12]. However, such strategies are not guaranteed to capture the intentions of the SO. Consider the following scenario: in a hospital, “deny override” is the default conflict resolution strategy for the access control system. At the beginning, the SO introduces a rule by which a doctor cannot read the medical record of patients that are not in the doctor’s ward. One year later, the SO introduces a rule that allows access of doctors on night duty to the medical records of all patients in the hospital, but she forgets to amend the earlier rule. Because of “deny override”, the later permission will be ignored by the system.

This example shows the need for “Policy Assistants” that interact with the SO when the policy set is modified. The Assistant would detect and signal inconsistencies at the time they are being created, and would prompt the Security Officer for manual resolution according to her intentions. In our example, an obvious resolution is the removal of the earlier rule. Work in this paper is meant to be a contribution towards the creation of such assistants.

Techniques for conflict detection have been less studied than techniques for conflict resolution. In [14] a graph-based approach has been used to resolve conflicts in context-aware access control policies. In [13], authors specify policies in a graph-based specification formalism and use formal properties of graph transformations to detect inconsistencies between access control rules. Furthermore, interesting methods and principles have been used for conflicts detection in firewall rules [2,3,6,7,10,15].

We investigate the conflict detection problem in a fairly general model taking into account different access control specification properties including abstract rules, and positive and negative authorizations as well as context expressions. We define a type system that enables us to check the specified access control rules for consistency.

The remaining of the paper is organized as follows. Section 2 describes a model for access control policies at different levels of abstraction specified with context expressions. A first order logic is used to express the security policy of the model, which handles abstract and concrete access control policies. Section 3 presents a method to generate dynamic groups according to the specified contexts. The dynamic groups are used as input to a typing system for conflict detection. Further, a typing system is presented, that is capable of detecting all conflicts. Section 4 presents concrete examples that consider policy specifications within the health-care sector. Finally, Section 5 concludes the paper and provides suggestions for further work.

2 Access Control Policies With Contexts

The main goal of access control policies consists in specifying the authorizations (permissions and prohibitions) that regulate the different actions that may be performed by subjects on objects. These authorizations may be expressed using first-order logic formulas. For instance, the predicate *permission* (s, a, o) (resp. *prohibition*(s, a, o)) expresses a fact meaning that a subject s is

permitted (resp. prohibited) to perform action a on object o , while predicate $permission(doctor, read, medical_record)$ means that any doctor may read any medical record. The need for additional expressiveness, as discussed in the introduction, leads to a rule-based language such as the one that was proposed in [9,11,12], where each authorization may be expressed as follows:

$$\forall s \in S, \forall a \in A, \forall o \in O, (Condition) \rightarrow authorization(s, a, o)$$

where authorization may be a permission (resp. prohibition), S is a set of subjects, A a set of actions and O a set of objects. We note that we consider a positive or a negative authorization as a authorization in our specification.

The above rule means that for any subject s , action a and object o , if the provided condition is satisfied, then subject s is permitted (resp. prohibited) to perform action a on object o . Notice that prohibition is the negation of permission; i.e. $\neg permission(s, a, o) \stackrel{\text{def}}{=} prohibition(s, a, o)$ meaning that the fact that a subject s is not permitted to perform action a on object o is equivalent to the fact that subject s is prohibited to perform action a on object o .

We can have different types of constraints, since they can involve subjects, actions, objects and various combinations of them. These constraints should be satisfied for applying the authorization. Each constraint is expressed as a logical expression. For instance, the following rule:

$$R : doctor(s) \wedge medical_record(o) \wedge identity_patient(o, p) \wedge \\ different_ward(s, p) \rightarrow prohibition(s, read, o)$$

states that a doctor is not allowed to read a medical record of a patient if she/he is not in the same ward. In this example, we have (1) a subject constraint corresponding to the predicate $doctor(s)$, meaning that subject s is a doctor, (2) an object constraint corresponding to the predicate $medical_record(o)$ meaning that object o is a medical record and (3) a subject-action-object constraint defined as follows: $identity_patient(o, p) \wedge different_ward(s, p)$, where (a) $identity_patient(o, p)$ is an application dependent predicate saying that object o is a medical record corresponding to patient p and (b) $different_ward(s, p)$ is an application dependent predicate stating that subject s and patient p are located in different wards.

While there are different models to express policies such as RBAC [16], OrBAC [1] etc., we focus our work on a new model that we call CA-BAC (Concrete and Abstract Based Access Control). This model specifies access control policies by considering two levels and is thus more expressive than others in common use. The first level is abstract and the second is concrete. In addition to this, we introduce within our model the notion of dynamic user groups, which make the specification more flexible for expressing high level access control policies. In the following, we briefly discuss the proposed CA-BAC model.

2.1 Expressing High Level Access Control Rules

Constraints over subjects, actions and objects are specified by means of the following predicates:

– U_group is a predicate defined over the domains $S \times UG$, where S is a set of subjects and UG a set of user groups. If s is a subject and ug a user_group, then $U_group(s, ug)$ means that subject s is assigned to user group ug .

– A_group is a predicate defined over domains $A \times AG$, where A is a set of actions and AG a set of activity groups. If a is an action and ag an activity_group, then $A_group(a, ag)$ means that action a is assigned to activity group ag .

– V_group is a predicate defined over domains $O \times VG$, where O is a set of objects and VG a set of view groups. If o is an object and vg a view_group, then $V_group(o, vg)$ means that object o is assigned to view group vg .

Constraints that combine subjects, actions, and objects are modeled using the notion of context. We note that our definition of the context is quite similar to that of the OrBAC model [1]. From now on, the context will be specified using the predicate $Occurs$ that is defined as follows:

– $Occurs$ is a predicate that is defined over $S \times A \times O \times C$, where C is a set of contexts. If s is a subject, a an action, o an object and c a context then $Occurs(s, a, o, c)$ specifies that context c is satisfied for subject s , action a and object o .

The conditions that should be satisfied in order to relate a context to a subject, action and object are expressed using logical rules. Section 2.3 presents different examples for such rules. For instance, a default context is defined when no condition should be satisfied to grant the corresponding authorization. This may be defined as follows:

$$\forall s \in S, \forall a \in A, \forall o \in O, Occurs(s, a, o, default).$$

As another example, $patient_doctor$ is a context that may be defined as follows¹:

$$\forall s \in S, \forall a \in A, \forall o \in O, Occurs(s, a, o, patient_doctor) \leftarrow patient(s, o)$$

The above specification means that context $patient_doctor$ is satisfied between subject s , action a and object o if o is a patient of doctor s .

Policy rules definition. As presented in Section 2, each rule is expressed as follows:

$$\forall s \in S, \forall a \in A, \forall o \in O, ((Condition) \rightarrow authorization(s, a, o))$$

Using the above expression, $Condition$ corresponds now to the following expression:

$$U_group(s, ug) \wedge A_group(a, ag) \wedge V_group(o, vg) \wedge Occurs(s, a, o, context)$$

As an example, we can specify that “a doctor can prescribe medicine to his patients” as follows:

¹ Following [12] and the subsequent litterature, we write $B \leftarrow A$ to mean that from A one can infer B .

$$U_group(s, doctor) \wedge A_group(a, prescribe) \wedge V_group(o, patient) \\ \wedge Occurs(s, a, o, patient_doctor) \rightarrow permitted(s, a, o)$$

However, we do not express authorizations directly on concrete subjects, objects and actions for specifying high level access control rules. In fact, we first specify a authorization (positive or negative authorization) between user groups, activity groups, view groups and contexts. This high level authorization is a relation that is defined over domains $UG \times AG \times VG \times C$. For instance, $Permission(ug, ag, vg, c)$ means that user group ug is granted the permission to perform activity group ag on view group vg within context c . For convenience and for differentiating between high level and concrete level authorizations, we use the relation $Permission$ for expressing high level permission and $permitted$ for expressing permission at the concrete level. The same relations are defined for negative authorizations (using the relation $Prohibition$).

Using these high level authorizations such as $permission$ (resp. $prohibition$), the concrete level authorization $permitted$ (resp. $prohibited$) is derived from the $permission$ (resp. $prohibition$) assigned to user groups, activity groups and view groups by the relation $permission$ (resp. $prohibition$). Now, we can specify such permission² policies as follows:

$$\forall s \in S, \forall a \in A, \forall o \in O, \forall ug \in UG, \forall ag \in AG, \forall vg \in VG, \forall c \in C \\ Permission(ug, ag, vg, c) \wedge U_group(s, ug) \wedge A_group(a, ag) \wedge \\ V_group(o, vg) \wedge Occurs(s, a, o, c) \\ \rightarrow permitted(s, a, o)$$

meaning that subject s is permitted to perform action a over object o if in context c (1) user group ug is granted the permission to perform activity ag on view vg , (2) s is assigned to user group ug , (3) a is assigned to activity group ag , (4) o is assigned to view group vg and (5) context c occurs between s , a and o .

Dynamic user, activity and view groups. When specifying high level policies, subjects are statically assigned to the predefined user groups. While this is useful to define static roles as in RBAC [16] or OrBAC [1], some other groups may be defined dynamically according to specific contexts. For instance, we may want to specify a rule policy that says that all subjects in the emergency ward can read all medical records and cannot prescribe medicine to patients. If we use the above high level security policy specifications, then we have to write as many high level policies as the number of predefined user groups. A dynamic user group is not defined statically but is dynamically activated using a context defined over the subject. Once the dynamic group is activated, subjects satisfying the corresponding context are automatically assigned to it. This is modeled using the predefined predicate $Occurs_dynamic_ugroup$.

² with the same syntax we derive concrete prohibitions (denoted by relation $prohibited$).

– *Occurs_dynamic_ugroup* is a predicate that is defined over domains $S \times SC$ (where SC is a set of contexts that are defined over domain S). If s is a subject and sc a context for subject s then *Occurs_dynamic_ugroup*(s, sc) specifies that context sc is satisfied over subject s .

Then the corresponding subject s is implicitly assigned to dynamic user group dug_{sc} as follows:

$$U_group(s, dug_{sc}) \leftarrow Occurs_dynamic_ugroup(s, sc)$$

We also define two other predicates *Occurs_dynamic_agroup* (resp. *Occurs_dynamic_vgroup*) for dynamically activating activity groups (resp. view groups):

– *Occurs_dynamic_agroup* is a predicate that is defined over domains $A \times AC$ (where AC is a set of contexts that are defined over domain A). If a is an action and ac a context over action a then *Occurs_dynamic_agroup*(a, ac) specifies that context ac is satisfied over action a .

The corresponding action a is implicitly assigned to dynamic activity group dag_{ac} as follows:

$$A_group(a, dag_{ac}) \leftarrow Occurs_dynamic_agroup(a, ac)$$

– *Occurs_dynamic_vgroup* is a predicate that is defined over domains $O \times VC$ (where OC is a set of contexts that are defined over domain O). If o is an object and oc a context over object o then *Occurs_dynamic_vgroup*(o, oc) specifies that context oc is satisfied over object o .

The corresponding object o is implicitly assigned to dynamic view group dvg_{oc} as follows:

$$V_group(o, dvg_{oc}) \leftarrow Occurs_dynamic_vgroup(o, oc)$$

Notice that $DUG \subseteq UG$ (resp. $DAG \subseteq AG$, $DVG \subseteq VG$) where DUG is the set of activated dynamic user groups (resp. DAG is the set of activated dynamic activity groups, DVG is the set of activated dynamic view groups) and UG the set of all user groups (i.e. activated dynamic user groups and statically defined user groups) (resp. AG the set of all activity groups and VG the set of all view groups).

2.2 Expressing Low Level Access Control Rules

Low level policies should be defined when there are authorizations that apply directly to subjects, actions and objects within a context. Of course, it is possible to define a user group (singleton user group) for which we assign only one subject. This solution is not interesting since it renders the model complex with useless subjects-user groups assignments (resp. actions-activity groups and objects-view groups). Our model proposes defining low level policies in addition to the high level ones specified above. Each access control policy rule is expressed as follows:

$$\begin{aligned} & \forall s \in S, \forall a \in A, \forall o \in O, \forall c \in C, \\ & Permission(s, a, o, c) \wedge Occurs(s, a, o, c) \\ & \rightarrow permitted(s, a, o) \end{aligned}$$

2.3 Expressing Contexts

The different authorizations apply when the corresponding constraints are satisfied. As we have seen in the previous section, the first three constraints correspond to separate conditions over subject, action and object. However, the last constraint in the condition part of the rule is expressed as a constraint over subject, action and object. This constraint corresponds to a set of elementary contexts that must be satisfied for applying the authorization. Each elementary context is defined over a subject, action and object. Our model allows specifying different types of contexts such as temporal, spatial, knowledge based and historical contexts.

– **Temporal context** that specifies the time constraint that must be satisfied for the subject to be granted with the requested access. To gain access, the current time should satisfy the temporal context. We consider that we have a trusted “*Clock*” that provides us with the accurate time. This clock may be requested at any time to provide the current time in order to assess the temporal context of the access control request. Other time values may be obtained from “*Clock*”: Time, Weekday, Monthday, Month, Monthweek, Yearweek. Two other basic functions, over the time set T , are used to express the temporal context: $start_time(t)$ and $end_time(t)$ where:

$$\begin{aligned} \forall s \in S, \forall a \in A, \forall o \in O \\ \forall t, t' \in T, Occurs(s, a, o, start_time(t)) \leftarrow Time(Clock, t') \wedge t' \geq t \\ \\ \forall s \in S, \forall a \in A, \forall o \in O \\ \forall t, t' \in T, Occurs(s, a, o, end_time(t)) \leftarrow Time(Clock, t') \wedge t' \leq t \end{aligned}$$

Using the above defined basic temporal contexts, we can define composed contexts that can be expressed by using different logical operators. For instance, let us consider the “*visitinghours*” context defined in the following security policy rule. Receptionists can locate patients during visiting hours where the visiting hours temporal context corresponds to the morning hours from 11h00 to 12h00 and only on the first two Mondays of the month. *visitinghours* temporal context is expressed as follows:

$$\begin{aligned} start_time(11h00) \wedge end_time(12h00) \wedge \\ on_weekday(monday) \wedge (on_monthweek(1) \vee on_monthweek(2)) \end{aligned}$$

– **Spatial context** corresponds to the spatial location constraints of the subject and object. This context defines the constraint, which depends on the subject and/or object location, that should be satisfied in order to grant the access authorization to the requested action. We assume that we have a trusted GPS system (or an access control system to the building and different places

within the building) that indicates the effective place of the subject or the object. Many spatial contexts may be defined. For instance, we may define a country, continent, town, street address, emergency ward of a hospital, etc. as a spatial context. We use different attributes for this context such as country, town, ward, street, etc. To specify that a subject s (or object o) is located in emergency ward, we use the predicate *is_located* to get this information from the *GPS* object.

A relation that is very useful in a hospital context is the relation specifying that the doctor and the patient are in the same ward. For example this context may be used to allow doctors prescribe medication to patients if they are in the same ward. In this example, *are_in_same_ward* might be defined as follows:

$$\begin{aligned} &\forall s \in S, \forall a \in A, \forall o \in O \\ &Occurs(s, a, o, are_in_same_ward) \\ &\leftarrow in_ward(s, w) \wedge in_ward(o, w) \end{aligned}$$

where $\begin{cases} in_ward(s, w) \leftarrow is_located(GPS, s, w) \\ in_ward(o, w) \leftarrow is_located(GPS, o, w) \end{cases}$

– **Knowledge based context** that depends on information that may be provided by the information system. In some circumstances, a request is granted according to some information stored in the information system database. For instance, a doctor can operate a patient only if he has at least 19 years of experience. The corresponding context *has_19_years_experience* may be expressed as follows:

$$\begin{aligned} &\forall s \in S, \forall a \in A, \forall o \in O \\ &Occurs(s, a, o, hasmorethan_19_years_experience) \\ &\leftarrow experience(s, years) \wedge years \geq 19 \end{aligned}$$

where *experience(s, years)* is a basic function that retrieves from the information system database the number of practice years of subject s .

– **Historical context** depends on the actions that are already performed. Some access requests could not be granted unless some actions are performed before the request is presented. A database logging the different actions (with the corresponding subjects, objects and timestamps) is used for this goal. For instance, a doctor cannot operate a patient unless he has already diagnosed him. The corresponding context *has_diagnosed* may be expressed as follows:

$$\begin{aligned} &\forall s \in S, \forall a \in A, \forall o \in O \\ &Occurs(s, a, o, has_diagnosed) \\ &\leftarrow log(s, diagnose, o) \end{aligned}$$

where $\log(s, \text{diagnose}, o)$ is a dependent predicate that says that action *diagnose* has already been performed by s over o . The different actions that are performed are stored in an event log database.

Notice that the context types are not limited to the above defined contexts. Others may be used including the different weather states (hot, cold, temperature, cloudy, windy, etc.), urgent cases when dealing with accidents in hospitals or threat context when dealing with intrusions in information systems, etc. Our objective is not to enumerate all possible contexts but to give an idea of contexts and how they are expressed for the goal of conflict detection.

2.4 User Group, Activity Group and View Group Hierarchies

We denote the hierarchy between user groups by using the following predicate $\text{usergroup_membership}(ug_1, ug_2)$ meaning that usergroup ug_1 is a *sub_usergroup* of ug_2 . Therefore, we get the following authorization inheritance according to the user group hierarchy:

$$\begin{aligned} &\forall ug \in UG, \forall ag \in AG, \forall vg \in VG, \\ &\text{usergroup_membership}(ug_1, ug_2) \wedge \text{authorization}(ug_2, ag, vg)) \\ &\rightarrow \text{authorization}(ug_1, ag, vg)) \end{aligned}$$

Accordingly, we respectively define the activity group and the view group hierarchies. The activity group hierarchy is defined using the predicate $\text{activitygroup_membership}(ag_1, ag_2)$ meaning that activity group ag_1 is a *sub_activity* of ag_2 . The view group hierarchy is defined by using the predicate $\text{viewgroup_membership}(vg_1, vg_2)$ meaning that view group vg_1 is a *sub_viewgroup* of vg_2 . The authorization inheritance according to the activity group and view group hierarchy are as follows:

$$\begin{aligned} &\forall ug \in UG, \forall ag \in AG, \forall vg \in VG, \\ &\text{activitygroup_membership}(ag_1, ag_2) \wedge \text{authorization}(ug, ag_2, vg)) \\ &\rightarrow \text{authorization}(ug, ag_1, vg)) \end{aligned}$$

$$\begin{aligned} &\forall ug \in UG, \forall ag \in AG, \forall vg \in VG, \\ &\text{viewgroup_membership}(vg_1, vg_2) \wedge \text{authorization}(ug, ag, vg_2)) \\ &\rightarrow \text{authorization}(ug, ag, vg_1)) \end{aligned}$$

3 Conflict Verification by Typing

Access control rules can be checked for several security properties such as consistency, completeness, redundancy, determinism, etc. In the following, we focus on the consistency property. A set of rules is consistent if no active entity (users or group of users) has both positive and negative authorizations to access a resource. An active entity can receive authorizations explicitly from a rule or from

rules that grant authorizations to a group to which this active entity belongs. To test this condition, a typing system is introduced which checks that there are no conflicting rules for any given entity. Our typing system manipulates judgments of the form $\Gamma \vdash_{UG} RG : \tau$ which can be read: in the environment Γ , the resource group RG has a type τ for the user group UG . The type τ is *ok* if there is no conflict for the user group UG in accessing RG , otherwise it is *fail*.

3.1 Dynamic Groups

In order to conduct our analysis, we first extrapolate the context from access rules by generating as many access rules as there are context combinations. This manipulation introduces the notion of dynamic groups (users, activities and views). Hence, we identify for each rule the different user groups (resp. activity groups and view groups) that satisfy the conditions for granting the corresponding rules authorizations. This is performed by instantiating contexts within the rules. For each context c we consider the two cases when it is satisfied or not (c and \bar{c}). For instance, let us consider the following rule: “*doctors in emergency ward are allowed to read all medical records*”. However according to other rules, not all doctors are allowed to read all medical records but only those that are in the emergency ward. Thus, we identify two groups of doctors under the “emergency” context.

For generating the dynamic groups, we choose to annotate groups with their context’s instantiation. For instance, let us consider the following rule: “*doctors may prescribe medication to their patients*”, which is expressed as follows:

$$\textit{Permission}(\textit{doctor}, \textit{prescribe}, \textit{patient}, \textit{patient_doctor})$$

We split the user group *doctor* into $\textit{doctor}_{\textit{patient_doctor}}$ and $\textit{doctor}_{\overline{\textit{patient_doctor}}}$. For each access control rule, each group is likely to be split into two dynamic groups representing those for which the context is satisfied and those for which it is not satisfied.

3.2 Typing System

The main purpose of our typing system is to verify that two user groups that have common elements should not have different access rights. If such a situation occurs, then it is possible that elements belonging to both groups are simultaneously permitted and prohibited to access a given resource, leading to a conflict.

3.3 Examples

Let us consider the example of rules in a hospital. We consider three different user groups, namely *doctor*, *nurse* and *chief*.

Assume *chief* user group is composed of two sub groups; (1) head doctor and (2) head nurse. Also assume that the following access control rules are part of the internal security policy of the hospital:

- (1) Doctors are not authorized to locate patients
- (2) Head_doctors can locate patients

In our access control model, these two rules are expressed as follows:

- (1) *prohibition(doctor, locate, patient, default)*
- (2) *permission(head_doctor, locate, patient, default)*

Notice that user group *head_doctor* is a *sub_user_group* of *doctor*. In our model, this is represented using hierarchy (Section 2.4), as follows:

$$\textit{usergroup_membership}(\textit{head_doctor}, \textit{doctor})$$

meaning that user group *head_doctor* inherits all authorizations of user_group *doctor*. From the first rule, we infer that head doctors are not authorized to locate patients since they inherit the prohibition assigned to doctors. Therefore, the head doctors are both allowed and denied to locate patients.

Other rules state that:

- Doctors can consult any patient’s medical record
- Nurses can’t consult a patient’s medical record if they are not assigned to the patient’s room

These authorizations are expressed in our model as follows:

- (1) *permission(doctor, consult, medical_record, default)*
- (2) *prohibition(nurse, consult, medical_record, are_in_same_ward)*

At a first glance, these two rules could not be in conflict because the corresponding user groups (*doctor* and *nurse*) are disjoint. However, this is not always the case since in some hospitals, there are some doctors that may play the role nurses meaning that they are assigned to *nurse* and *doctor* user groups.

We present in the following sections the typing system that is used for detecting the different conflicts. This typing system checks that non disjoint user groups do not have both permission and prohibition authorization over a common object.

3.4 Typing Judgements and Typing Rules

We use the typing relations \vdash_{UG} for groups of users *UG*. Each resource is typed for all user groups. The typing environment containing actions and their corresponding authorizations. For instance $\{(read, deny), (write, permit)\}$ represents an environment.

We define two kinds of typing judgements $\Gamma \vdash \diamond$ denoting a well typed environment and $\Gamma \vdash_{UG} RG : \tau$ denoting the type of the resources *RG* w.r.t group *UG*. This type may be either “*ok*” (no conflict) or “*fail*” (when conflict is present).

Table 1. Typing rules

$\frac{\square}{\phi \vdash \diamond}$	Empty environment
$\frac{\Gamma \vdash \diamond \quad a \notin \text{dom}(\Gamma)}{\Gamma \cup \{(a, \tau)\} \vdash \diamond}$	Add action
$\frac{\square}{\phi \vdash_{UG} RG : ok}$	Default judgment
$\frac{\Gamma \vdash_{UG} RG : ok \quad \langle UG', RG', \tau, a \rangle \quad UG' \cap UG \neq \phi \quad RG \cap RG' \neq \phi \quad a \notin \text{dom}(\Gamma)}{\Gamma \cup \{(a, \tau)\} \vdash_{UG' \cap UG} RG \cap RG' : ok}$	Acquisition 1
$\frac{\Gamma \vdash_{UG} RG : ok \quad \langle UG', RG', \tau, a \rangle \quad UG' \cap UG \neq \phi \quad RG \cap RG' \neq \phi \quad (a, \tau) \in \Gamma}{\Gamma \vdash_{UG' \cap UG} RG \cap RG' : ok}$	Acquisition 2
$\frac{\Gamma \vdash_{UG} RG : ok \quad \langle UG', RG', \tau, a \rangle \quad UG' \cap UG \neq \phi \quad RG \cap RG' \neq \phi \quad (a, \bar{\tau}) \in \Gamma}{\Gamma \cup \{(a, \bar{\tau})\} \vdash_{UG' \cap UG} RG \cap RG' : fail}$	Conflict 1
$\frac{\Gamma \vdash_{UG} RG : ok \quad \Gamma' \vdash_{UG'} RG' : ok \quad UG' \cap UG \neq \phi \quad RG \cap RG' \neq \phi \quad \exists a : \Gamma(a) \neq \Gamma'(a)}{\Gamma \vdash_{UG' \cap UG} RG \cap RG' : fail}$	Conflict 2

The typing rules are shown in Table 1. At the beginning of the verification, all resource groups are of type “ok” as a default judgment for all user groups. The *Acquisition 1* and *Acquisition 2* rules translate control rules into judgments. Rules *Conflict 1* and *Conflict 2* capture conflict situations: an action cannot take two contradictory types permit and deny for a user and a given resource. Rule *Conflict 2* is used to ensure compositionality of the typing system.

4 Example

To illustrate our method, we consider a set of policy rules deployed in a hospital where four distinct user groups are defined, namely *doctor*, *nurse* and *chief*. According to the security policy of the hospital, *doctor* and *nurse* groups are disjoint. However, *chief* and *nurse* groups are not disjoint. Furthermore, access control rules of the hospital are as follows:

- R1: Doctors have read/write access to their patient’s medical record
- R2: Doctors in the same ward as Patients has read access to the patient’s medical records

- R3: Chiefs have read access to all medical records
- R4: Nurses cannot read the patient’s medical record if they are not assigned to the patient’s ward

At the beginning, we extrapolate the context from rules R1, R2 and R4 as follows:

- R1: Doctors have read/write access to their patient’s medical record

The context $c1$ of the original rule $R1$ is defined as $c1 = patient_doctor$. As a consequence, by instantiating this context we generate two dynamic groups $doctor_{c1}$ and $doctor_{\overline{c1}}$. Similarly, we generate two dynamic resource groups $medical_record_{c1}$ and $medical_record_{\overline{c1}}$.

- R2: patient’s ward doctor has read access to the patient’s medical record

The corresponding context of this rule is $c2 = are_in_same_ward$ (see Section 2.3 on how to specify these contexts). The user group $doctor$ depends on contexts $c1$ and $c2$. We generate four dynamic user groups $doctor_{c1,c2}$, $doctor_{c1,\overline{c2}}$, $doctor_{\overline{c1},c2}$ and $doctor_{\overline{c1},\overline{c2}}$. Again, we generate four different dynamic resource group $medical_record_{c1,c2}$, $medical_record_{c1,\overline{c2}}$, $medical_record_{\overline{c1},c2}$ and $medical_record_{\overline{c1},\overline{c2}}$.

- R4: Nurses can’t read a patient’s medical record if they are not assigned to the patient’s ward. We generate from user group $nurse$ two dynamic groups $nurse_{c2}$ and $nurse_{\overline{c2}}$ and two dynamic resource groups $medical_record_{c2}$ and $medical_record_{\overline{c2}}$.

The result of this extrapolation process is the following six rules:

- R11: $doctor_{c1,c2}$ has read/write access to $medical_record_{c1,c2}$
- R12: $doctor_{c1,\overline{c2}}$ has read/write access to $medical_record_{c1,\overline{c2}}$
- R21: $doctor_{c1,c2}$ has read access to $medical_record_{c1,c2}$
- R22: $doctor_{\overline{c1},c2}$ has read access to $medical_record_{\overline{c1},c2}$
- R31: Chief has read access to all medical records
- R41: $nurse_{\overline{c2}}$ can’t read $medical_record_{\overline{c2}}$.

By applying our typing system, we find an inconsistency in this access control example, which arises if there are nurses that are also chiefs. The proof is presented in Table 2.

5 Conclusion

In the first part of this paper, we have presented a model that can be used in order to specify access control policies both at the abstract and at the concrete levels, by considering contexts. This is an innovation with respect to existing

Table 2. Tree Proof

$\frac{\square}{\phi \vdash_{chief} \text{medical_record}: ok}$	(Default judgement)
$\frac{\begin{array}{c} \langle chief, \text{medical_record}, \text{permit}, \text{read} \rangle \\ \text{read} \notin \text{dom}(\emptyset) \text{ write} \notin \text{dom}(\emptyset) \end{array}}{\{(read, \text{permit})\} \vdash_{chief} \text{medical_record}: OK}$	(Acquisition 1)
$\frac{\begin{array}{c} \langle Nurse_{c_2}, \text{medical_record}_{c_2}, \text{deny}, \text{read} \rangle \\ Nurse_{c_2} \cap Chief \neq \emptyset \text{ medical_record}_{c_2} \cap \text{medical_record} \neq \emptyset \end{array}}{\{(read, \text{permit}), (read, \text{deny})\} \vdash_{Chief \cap Nurse_{c_2}} \text{medical_record} \cap \text{medical_record}_{c_2}: fail}$	(Conflict 1)

access control models, where complete flexibility of expressing all elements of policies (subject, action and object) at different levels of abstraction does not exist. As in every access control system, conflicts between rules are possible. Such conflicts can be unintentionally introduced by security officers when they update policies. In order to detect them, we propose a typing system that is applied to the set of rules, and will yield a verdict of “conflict” or “no conflict”. To our knowledge, no similar typing system, that considers contexts, is available in the literature.

In this paper, we only consider positive and negative authorizations. In our future work, we will consider other decisions such as obligations for which some actions should be launched once certain conditions are satisfied. Such decisions will lead to more complex conflict situations. The other issue we are starting to investigate is delegation that may generate conflicts with other rules.

Acknowledgments

This research has been funded in part by grants from the Natural Sciences and Engineering Research Council of Canada and from CA Labs. The authors wish to thank Nadera Slimani for many research discussions.

References

1. AbouElKalam, A., El Baida, R., Balbiani, P., Benferhat, S., Cuppens, F., Deswarte, Y., Miège, A., Saurel, C., Trouessin, G.: Organization Based Access Control. In: Proceedings of IEEE 4th International Workshop on Policies for Distributed Systems and Networks (POLICY 2003), Lake Como, Italy, June 2003, pp. 120–134 (2003)
2. Adi, K., Elkabbal, A., Mejri, M.: Un Système de Types pour l’Analyse des Pare-feux. In: Proceedings of the 4th Conference on Security and Network Architectures (SAR 2005), pp. 227–236 (2005)

3. Al-Shaer, E., Hamed, H., Boutaba, R., Hasan, M.: Conflict classification and analysis of distributed firewall policies. *IEEE Journal on Selected Areas in Communications* 23(10), 2069–2084 (2005)
4. Bertino, E., Catania, B., Ferrari, E., Perlasca, P.: A logical framework for reasoning about access control models. *ACM Trans. Inf. Syst. Secur.* 6(1) (2003)
5. Bertino, E., Jajodia, S., Samarati, P.: Supporting Multiple Access Control Policies in Database Systems. In: *IEEE Symposium on Security and Privacy*, pp. 94–107 (1996)
6. Bouzida, Y.: Managing security rules conflicts. European Patent Number EP 2 023 567 A1 (August 2007)
7. Bouzida, Y.: Online security rules conflict management. European Patent Number EP 2 023 566 A1 (August 2007)
8. Cuppens, F., Cuppens-Boulahia, N., BenGhorbel, M.: High Level Conflict Management Strategies in Advanced Access Control Models. *Electr. Notes Theor. Comput. Sci.* 186, 3–26 (2007)
9. Cuppens, F., Miège, A.: Modelling contexts in the Or-BAC model. In: *Proceedings of the 19th Annual Computer Security Applications Conference (ACSAC 2003)*, Las Vegas, Nevada, USA, December 2003, pp. 416–427 (2003)
10. Gouda, M.G., Liu, A.X.: Firewall Design: Consistency, Completeness, and Compactness. In: *ICDCS 2004*, pp. 320–327 (2004)
11. Weissman, V., Halpern, J.Y.: Using First-Order Logic to Reason about Policies. In: *16th IEEE Computer Security Foundations Workshop, CSFW 2003* (2003)
12. Jajodia, S., Samarati, P., Subrahmanian, V.S.: A logical language for expressing authorizations. In: *IEEE Symposium on Security and Privacy*, pp. 31–42 (1997)
13. Koch, M., Mancini, L., Parisi-Presicce, F.: Conflict detection and resolution in access control policy specifications. In: Nielsen, M., Engberg, U. (eds.) *FOSSACS 2002*. LNCS, vol. 2303, pp. 223–237. Springer, Heidelberg (2002)
14. Masoumzadeh, A., Amini, M., Jalili, R.: Conflict detection and resolution in context-aware authorization. In: *AINAW 2007: Proceedings of the 21st International Conference on Advanced Information Networking and Applications Workshops*, pp. 505–511. IEEE Computer Society, Los Alamitos (2007)
15. Pene, L., Adi, K.: Calculus for Distributed Firewall Specification and Verification. In: *Proceedings of 5th International Conference on Software Methodologies, Tools and Techniques*, pp. 301–315. IOS Press, Amsterdam (2006)
16. Sandhu, R.S., Coyne, E.J., Feinstein, H.L., Youman, C.E.: Role-based access control models. *IEEE Computer* 29(2), 38–47 (1996)

Dynamic Pricing in Electronic Commerce Using Neural Network

Tapu Kumar Ghose and Thomas T. Tran

School of Information Technology and Engineering,
University of Ottawa, Ottawa, ON K1N 6N5, Canada
{tghos009,ttran}@site.uottawa.ca

Abstract. In this paper, we propose an approach where feed-forward neural network is used for dynamically calculating a competitive price of a product in order to maximize sellers' revenue. In the approach we considered that along with product price other attributes such as product quality, delivery time, after sales service and seller's reputation contribute in consumers purchase decision. We showed that once the sellers, by using their limited prior knowledge, set an initial price of a product our model adjusts the price automatically with the help of neural network so that sellers' revenue is maximized.

Keywords: Electronic Commerce, Dynamic Pricing, Neural Network.

1 Introduction

There exist intelligent agents which enable online sellers to dynamically calculate a competitive price for a product. Among them, some intelligent agents assume that sellers are provided with complete knowledge of market parameters, while some other agents consider product price as the only attribute that determines consumers' purchase decision [1]. In recent decades extensive research has been done in dynamic pricing. Some of the research made an assumption that there is only one seller in the market [8]. However, in real life sellers have limited or no prior knowledge about the market parameters (e.g., buyer's reservation price, competitive sellers' price and profit etc). In addition, in reality there exist several competitive sellers in online market. Moreover, micro-economic literature and online consumer surveys suggest that a consumer's purchase decision is determined by multiple product attributes including price, delivery time, seller reputation, product quality and after-sale service [2]. In this paper we attempt to address the problem of dynamic pricing in a competitive online economy, where a buyer's purchase decision is determined by multiple attributes. In our model we consider five attributes. They are product price, product quality, delivery time, after-sale service, and sellers' reputation. However, our model is general enough to work for any number of attributes. We use feed-forward neural network to determine an optimal price for the products in order to maximize sellers' revenue. In our simulation we showed that once the sellers set an initial price of the product, our model adjusts the price of the product automatically with the

help of neural network in order to maximize profits. In setting the initial price of a product, we assume that sellers use their prior knowledge about the prices of the product offered by other competing sellers.

The remaining of the paper is organized as follows: Section 2 provides background information on feed-forward neural network. Section 3 presents our proposed approach for dynamic pricing. Section 4 represents results and analysis from our simulation. Section 5 discusses related work. Section 6 concludes the paper with future research directions.

2 Feed-Forward Neural Network

In feed-forward network each unit in a specific layer receives input only from the units in the immediately preceding layer. Each unit u_i has an activation value a_i which acts as output of the unit.

$$a_i = f \left(\sum_{j=0}^{i-1} W_{j,i} a_j \right). \quad (1)$$

where $\sum_{j=0}^{i-1} W_{j,i} a_j$ is the weighted sum of the inputs to unit u_i and f is the activation function applied to the weighted sum. In our model we have chosen logistic sigmoid function as the activation function.

We used a bias unit which is connected to unit u_0 of the input layer. We set the production cost of the product as the output of the bias unit. In addition, we set the numerical weight of the links associated with the bias unit to 1. With the help of the additional bias unit we can ensure that our network will never provide the price of a product below its production cost.

3 Dynamic Pricing Using Neural Network

In our model we are considering five attributes which contribute in buyers' purchase decisions. We are assigning the price of a product where buyers preferred attributes are product price, product quality, delivery time, after sale service, and sellers' reputation to u_1, u_2, u_3, u_4 and u_5 respectively (Fig. 1). The input layer also consists of one extra unit u_0 as the bias unit. We set the value of a_0 to the production cost of the product. Initially, all the values a_1, a_2, a_3, a_4 and a_5 of the input units are set by the sellers. In setting the initial price of a product, we assume that sellers use their prior knowledge about the prices of the product offered by other competing sellers. Our simulation showed better performance for three hidden units. Hence, in our model we used three units in the hidden layer. The price of the product determined by the network (Fig. 1) can be found by using final output a_9 . The value of a_9 can be calculated with the aid of equation (2) as follows:

$$Finaloutput, a_9 = f(W_{6,9}a_6 + W_{7,9}a_7 + W_{8,9}a_8). \quad (2)$$

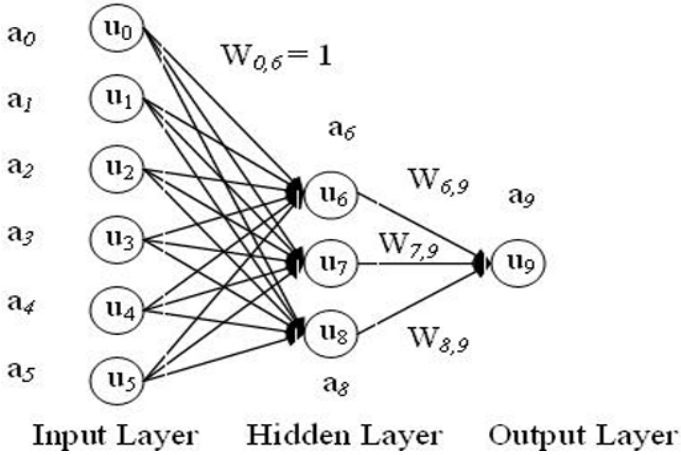


Fig. 1. A three-layered Feed-forward Neural Network

$$\begin{aligned}
 \text{where, } a_6 &= f(W_{0,6}a_0 + W_{1,6}a_1 + W_{2,6}a_2 + W_{3,6}a_3 + W_{4,6}a_4 + W_{5,6}a_5) \\
 a_7 &= f(W_{0,7}a_0 + W_{1,7}a_1 + W_{2,7}a_2 + W_{3,7}a_3 + W_{4,7}a_4 + W_{5,7}a_5) \\
 a_8 &= f(W_{0,8}a_0 + W_{1,8}a_1 + W_{2,8}a_2 + W_{3,8}a_3 + W_{4,8}a_4 + W_{5,8}a_5)
 \end{aligned}$$

Initially, we assume that the buyers have equal preference on all the five attributes that we are considering. Therefore, we associate each link between the layers with equal weights. The following is the process of dynamic pricing by neural network:

1. Input the five attributes (as mentioned earlier) of the product as the units of the input layer in the network.
2. Set the production cost of the product as the input to bias unit and set the weights of the links associated with bias unit to 1.
3. Sellers set the values (a_i) of the input units by using their prior knowledge about the prices of product offered by other competing sellers.
4. Associate the links between input units and hidden units with equal weight, i.e., 0.2. Also, associate the links between hidden units and output unit equally, i.e., 0.33.
5. Determine the price from the output layer.
6. Compute the actual revenue using the price determined by the networks.
7. Compute the error, i.e., the difference between the desired revenue and actual revenue.
8. If the error is approximately zero then go to step 11.
9. Update the weights of the links using back-propagation technique to minimize the error.
10. Go to step 5.
11. Set the price from the output layer as the product price.

Initially, the error size may be large depending on how the initial weights of the links and the values of the input units are chosen. The error is minimized at each iteration from step 5 through step 10. Once the price of a specific product is determined from the output layer from step 11, the weights of the links remain unchanged. The number of buyers purchasing the product at the determined price (say N) is then calculated. If N is greater than the expected number of buyers, then in step 4 instead of taking 0.2 as the weight $W_{i,j}$ between the input units and the hidden units, we use the weights, $W_{i,j}$, that were determined during the last iteration and go through the process again. For instance, assume that the value of $W_{1,6}$ was 0.38 when the product price was determined from step 11. In such scenario, we would like to set the value of $W_{1,6}$ to 0.38 instead of 0.2 in step 4 and run the process again.

On the other hand, if N is less than the expected number of buyers then the sellers' desired revenue is reduced by a small amount and entire process is run from beginning. The seller can increase their desired revenue in order to determine new price for the product.

4 Results and Analysis

Our network is run with five input units, three hidden units and one output unit. We started our simulation by training the network with 10 sets of training patterns¹ for five different epochs² or iterations. As the training continues, after each iteration in an epoch, the network calculates amount of error. The calculated error is then used to update the weights of the links by using back propagation algorithm. We run our network with learning rates of 0.01, 0.001 and 0.0001. The five different epochs used are 10, 100, 1000, 10000 and 100000. We let our network to tolerate an error of amount 0.01 and 0.001.

After finishing the training, we used our trained network to determine the price of a product (lets call it P). We considered 65.57 as the production cost of P . Since our model requires initial selling price of the product to be set by the seller, we set the initial selling price of P to 66.5, 65.8, 69.9, 69.2 and 68.4 where buyer's preferred attributes are product price, product quality, delivery time, after sale service, and sellers' reputation respectively.

The following two graphs (Fig. 2) show the amount of revenue earned after selling each product P to a single buyer at the determined price given by the network. The first graph shows the output where our network tolerated 0.01 amount of error, while the following graph is for 0.001 amount of error tolerance by the network.

From the result we can see that the amount of revenue earned per product after selling it to a single buyer is increased with the higher number of epochs used in training the network. However, the execution time is too long if number of epochs was chosen more than 10^5 . Hence, we run our simulation up to

¹ A training pattern consists of a set of inputs with desired output.

² Epoch is the maximum number of times the complete data set of training pattern can be used by the network for training.

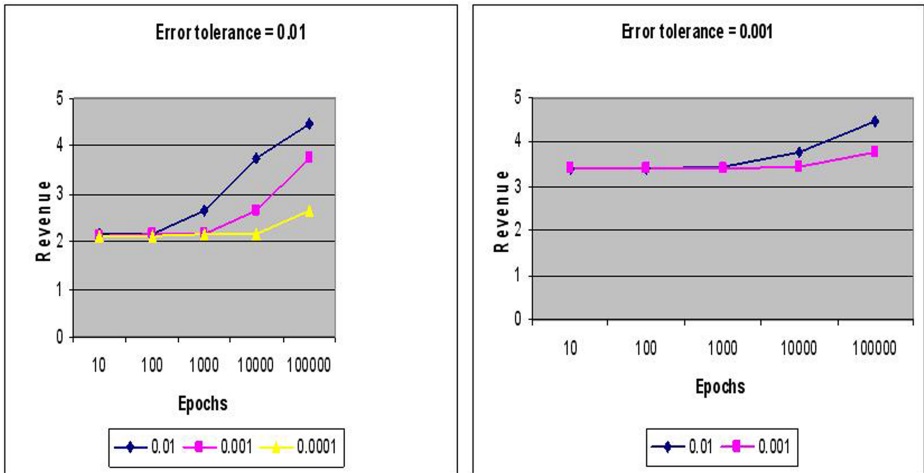


Fig. 2. Network performance in terms of Revenue earned

10^5 epochs. The graphs also demonstrate that the network performs best when 0.01 is used as learning rate.

5 Related Works

Over the past few years online dynamic pricing has stimulated considerable interest in commercial and research communities. Several analytical models have been developed for dynamic pricing in online economies [1,5,7]. C. Brooks et al used neural network for dynamic pricing where a monopolist market has been considered [3,4]. In contrast, our model is general enough to work for both monopolist market and a competitive market with multiple sellers. Chinthalapati et al [5] considered time and price attributes of the product in determining customers' buying decisions. They used reinforcement learning as a tool to study price dynamics in an electronic retail market consisting of two competitive sellers (multi-seller environments) and price sensitive and lead time sensitive customers. On the other hand, we are considering three more attributes (product quality, after sale service and sellers' reputation) other than time and price. In fact our model is general enough to work for any number of attributes. Dasgupta et al. [6] used derivative following strategy in determining dynamic pricing in a multi-agent economy where each seller has limited information about competitor's price. Kong [7] examined seller strategies for dynamic pricing in a market for which a seller has finite time horizon to sell its inventory. For this purpose, a dynamic pricing strategy is developed using neural network based on online learning (called SDNN strategy, Sales-Directed Neural Network). The discussed SDNN strategy takes in account the dynamics and resulting uncertainties of the market place. Here, the sellers have to figure out the demand curve of the

products. In contrary, our model of dynamic pricing does not require the sellers to figure out the demand curve of the products.

6 Conclusion and Future Work

The proposed approach described here used feed-forward neural network to determine product price dynamically. We used back propagation algorithm to minimize the errors while training the network with 10 training patterns. We considered buyer preferences over multiple product attributes. Our model is general enough to work for any number of attributes. One additional unit in the input layer needs to be added for each new attribute. On the other hand, in order to remove an attribute from the network the corresponding unit from the input layer, along with all the links that are connected to the unit, has to be eliminated. We also considered that sellers have limited prior knowledge about market parameters like how other competing sellers set the prices. Our model aids the sellers of competitive market in the automation of determining the price of a product in order to maximize their revenue. We would like to compare our approach with few other well known existing approach of dynamic pricing, like game-theoretic (GT), my-optimal (MY), derivative following (DF) etc. We are planning to compare the total revenue earned by different sellers after selling the same product whose price is determined by different strategies (GT, MY, DF and our designed approach). In the comparison we will be considering each seller follows different pricing strategies.

References

1. Dasgupta, P., Hashimoto, Y.: Multi-attribute dynamic pricing for online markets using intelligent agents. In: AAMAS (2004)
2. Brown, J., Goolsbee, A.: Does the Internet Make Markets More Competitive? In: NBER Working Papers 7996, National Bureau of Economic Research (2000)
3. Brooks, C., Gazzale, R., MacKie-Mason, J., Durfee, E.: Improving learning performance by applying economic knowledge. In: Proc. of the 3rd ACM Conference on Electronic Commerce (2003)
4. Brooks, C., Fay, S., Das, R., MacKie-Mason, J., Kephart, J., Durfee, E.: Automated strategy searches in an electronic goods market: learning and complex price schedules. In: Proc. of 1st ACM Conference on E-Commerce (1999)
5. Raju Chinthalapati, V.L., Yadati, N., Karumanchi, R.: Learning Dynamic Prices in MultiSeller Electronic Retail Markets With Price Sensitive Customers, Stochastic Demands, and Inventory Replenishments. IEEE (2006)
6. Dasgupta, P., Das, R.: Dynamic Service Pricing for Brokers in a Multi-Agent Economy. IEEE (2000)
7. Kong, D.: One Dynamic Pricing Strategy in Agent Economy Using Neural Network Based on Online Learning. In: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (2004)
8. Gallego, G., van Ryzin, G.: Optimal dynamic pricing of inventories with stochastic demand over finite horizons. *Manage. Sci.* 40(8), 999–1020 (1994)

TwoStep: An Authentication Method Combining Text and Graphical Passwords

P.C. van Oorschot and Tao Wan*

School of Computer Science, Carleton University, Ottawa, Canada
{paulv, twan}@scs.carleton.ca

Abstract. Text-based passwords alone are subject to dictionary attacks as users tend to choose weak passwords in favor of memorability, as well as phishing attacks. Many recognition-based graphical password schemes alone, in order to offer sufficient security, require a number of rounds of verification, introducing usability issues. We suggest a hybrid user authentication approach combining text passwords, recognition-based graphical passwords, and a two-step process, to provide increased security with fewer rounds than such graphical passwords alone. A variation of this two-step authentication method, which we have implemented and deployed, is in use in the real world.

Keywords: Graphical Passwords, User Authentication, Phishing, Security.

1 Introduction

Text passwords have been widely used for user authentication, e.g., by almost all websites on the Internet. However, it is well-known that text passwords are insecure for a variety of reasons. For example, users tend to choose simple passwords in favour of memorability, making them subject to dictionary attacks; and text passwords can be stolen by malicious software (e.g., keystroke loggers) when being entered from keyboards. Phishing is another serious threat to text passwords, by which, a user could be persuaded to visit a forged website and enter their passwords. Such an attack is made possible in part due to the fact that text passwords do not allow users to authenticate a server; by design they provide only one-way user authentication, and server authentication is not a design objective of text passwords alone.

We propose a two-step authentication method to strengthen text passwords by combining them with graphical passwords. In this approach, called *TwoStep*, users continue to use text passwords as a first step, but then must also enter a graphical password, providing the following advantages: (1) users' current sign-in experience is largely preserved; (2) a text password alone which is stolen (e.g., by phishing) does not compromise an account; (3) users can be alerted if not seeing the graphical password image after providing their text passwords, implicitly providing server authentication; and (4) it can be implemented in software alone, increasing the potential for large-scale adoption on the Internet.

* Corresponding author.

The rest of this paper is organized as follows. In Section 2, we describe TwoStep, and consider its security. Section 3 provides preliminary security analysis for TwoStep. Section 4 briefly reviews related work. We conclude in Section 5.

2 Two-Step Authentication Method

Given that text passwords are easy to deploy and to use, we believe that they will continue to be popular. Thus, we suggest that effort should be made to enhance text passwords with an easy to use additional defense mechanism that can address common password attacks, such as brute-force and phishing attacks. To this end, we propose *TwoStep*, a combination of text passwords and recognition-based graphical passwords. The latter can complement text passwords being less subjective to phishing attacks which require prior knowledge of users' image portfolios, and to naive keylogger attacks.

In step one, a user is asked for her user name and text password. After supplying this, and independent of whether or not it is correct, in step two, the user is presented with an image portfolio. The user must correctly select all images (one or more) pre-registered for this account in each round of graphical password verification. Otherwise, account access is denied despite a valid text password. Using text passwords in step one preserves the existing user sign-in experience. If the user's text password or graphical password is correct, the image portfolios presented are those as defined during password creation. Otherwise, the image portfolios (including their layout dimensions) presented in first and a next round are random but respectively a deterministic function of the user name and text password string entered, and the images selected in the previous round. More specifically, the image portfolio in round n is pseudo-randomly generated from a seed value derived from the entered user name and text password when $n = 1$, and from the images selected in round $n - 1$ when $n \geq 2$.

Seeing a portfolio including no familiar image allows a legitimate user to immediately realize that she entered an invalid text or graphical password (and then go back to re-enter it, e.g., using a "Go Back" dialog button), but prevents an attacker from knowing that the text or graphical password tried is invalid (cf. [32]).

Creation of Graphical Passwords. Graphical passwords can be created during user registration or after registration (for users registered before TwoStep was implemented), and be changed any time after creation. A graphical password policy, which may be set by the site operator or the user, influences its presentation and security. Example policy attributes are: *number of rounds of verification*; *display layout*, e.g., 6×6 , defining how images are presented to the user, and the total number of images displayed in each round; *number of images* to be selected in each round; and *ordered* or *unordered* image selection, defining whether order of image selection matters.

After a graphical password policy is defined, users choose images as their graphical passwords. For each round of verification, the specified number of images are randomly selected by the system from a database to form an image portfolio. A user then chooses a specified number of images from the portfolio as her graphical password components. This process repeats for the specified number of rounds. If the user does not like a particular image portfolio, she may request a new one or upload her own images to be included in a portfolio. An accepted image portfolio remains unchanged until the user

changes her graphical password. To facilitate recognition, images within a portfolio are assembled to be sufficiently distinguishable.

Subsequent Login Using TwoStep. In step one, the user as usual enters a user name and text password. The login page of the server deploying TwoStep remains the same as when text passwords alone were used, i.e., no change in the front login page is required to deploy TwoStep, nor do users see any difference in their sign-in experience in step one. After the user provides a text password, the second step of authentication (the graphical step or g-step) begins. In each round of graphical password verification, the server transmits an image portfolio to the user, and the user chooses out her pre-registered images. After the user completes all rounds of verification, if both the text password and all graphical passwords were correct, she is granted account access. Otherwise, access is denied. We next discuss several attacks against graphical passwords which must be considered. Further security discussion is found in Section 3.

Eavesdropping. An attacker able to intercept communication between the server and client would be able to capture image portfolios transmitted from the server, and the images selected by the user, thus stealing the entire graphical password. To prevent this attack, a security protocol such as HTTPS must be deployed to provide confidentiality.

Shoulder-Surfing. An attacker can also steal a graphical password by shoulder-surfing (e.g., using a video camera) during the g-step. Such shoulder-surfing would be particularly easy if an implementation of the g-step provided user visual feedback upon user selection of an image, such as highlighting an image border. Here we describe a simple method to mitigate this type of attack (see Fig. 1).

For a given image portfolio, each image is associated with an index number. Images along with their index numbers are displayed in a random order on the screen. Below the displayed image portfolio is a *selection panel* with all index numbers displayed incrementally. To select an image, the user identifies the image and then clicks the corresponding index number on the lower selection panel. In the case that several images must be chosen from a portfolio, the selection panel can help the user keep track of which images have been selected so far (and allows easy de-selection, by clicking the corresponding number in the bottom panel, if necessary). The idea is that it is more difficult for a casual human observer to have line of sight to the lower panel and to map an index or set of indices from it to the corresponding images on the screen. This approach can reduce casual shoulder-surfing

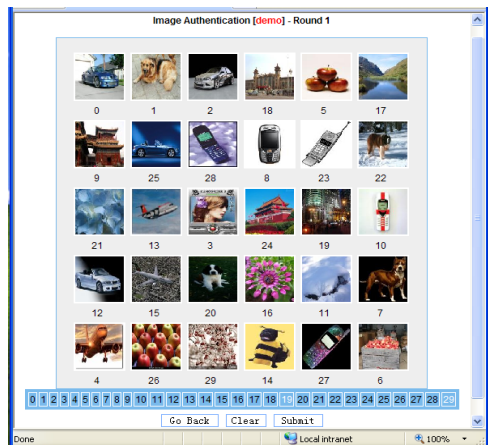


Fig. 1. Selection panel in graphical step

but cannot fully prevent such attacks involving movie-clip camera phones. Other techniques, e.g., Gaze-based password entry [13], can better mitigate this type of attack, but have their own usability and deployment challenges.

3 Preliminary Security Analysis

Password Strength. We discuss the strength of TwoStep, measured by entropy in bits, by considering both the entropy of the text password and the graphical password parts. A text password of length l characters has entropy of $l \cdot \log_2 c$ bits if characters are selected uniformly at random and independently from an alphabet of c characters. For example, a randomly generated 8-character password consisting of digits, lowercase, and uppercase has $8 \cdot \log_2 62 = 47.6$ bits of entropy.

Let r be the number of rounds of our graphical password verification. For each round, let n be the size of the image portfolio, and $k < n$ the total number of images selected from the portfolio as the graphical password. The entropy of a randomly selected graphical password conforming to this policy is $r \cdot \log_2 t$, where $t = \binom{n}{k}$, and $\frac{n!}{(n-k)!}$ for un-ordered and ordered images respectively.

As an example, consider $r = 1$, $n = 36$, $k = 3$, and unordered image selection, meaning one round of verification by selecting 3 images in any order from a portfolio of size 36. The entropy is $\log_2 \binom{36}{3} \approx 12.8$ bits. For $r = 2$, in theory this doubles to 25.6 bits, though in practice we might expect less unpredictability due to patterns in user choice [5]. Choosing different parameters k, n, r, t can increase security, but also changes usability. In addition, password guessing attacks in TwoStep must be done online (interacting with the server), which is more costly than offline attacks.

Note that text passwords used in practice are generally far from randomly and independently selected, and often lowercase only (cf. [10]), decreasing entropy. For example, an 8-character lowercase password has entropy about 37.6 bits if all characters were selected randomly and independently. But in practice, they perhaps have only 20-35 bits on average and less for some subsets of users. Relative to this more realistic estimate, the 25.6 bits (or even 12.8 bits) of added security from the graphical part is quite significant, against both targeted single-account exhaustive attacks, and system-wide multi-account attacks that might attempt as few as 3-5 guesses per account.

Mitigating Naive Keylogging Attacks. Keylogging is a common method for stealing user text passwords. A keylogger is malicious software which intercepts keystrokes on an infected machine as a user types. For example, Microsoft Windows provides (un-documented) interfaces facilitating interception of system events including keystrokes. With TwoStep, a user would use the keyboard for the text password part, and mouse clicks for the graphical parts. Thus, a naive keylogger cannot obtain the graphical parts. More sophisticated malware can capture both user screen contents and mouse clicks to recover a graphical password, with more effort.

Mitigating Phishing Attacks. Phishing [7] is another common technique for stealing passwords by fooling users to enter such information into a fraudulent website spoofing a legitimate one (e.g., a bank site). Social engineering tactics are often used (e.g., “urgent account update”, requests to verify fake transactions, etc.). In TwoStep, while

users' text password part can still be stolen by phishing, obtaining their graphical password parts is more difficult: without knowledge of users' image profiles, the phisher does not know what images to present in order to extract a graphical password.

Mitigating Active MITM Attacks. An active man-in-the-middle (MITM) attack allows an attacker to become an intervening proxy and control all communication between the user and the website (cf. [9]). SSL cannot mitigate this attack since an attacker can use SSL on both communication segments individually, so users (and end website) appear to be "operating securely". The proxy can be either malware on a user's local machine or located on a remote server (controlled by an attacker) to which the user is drawn by phishing techniques. Such an attacker can gain access to any information exchanged between a user and a website, thus can defeat TwoStep.

It appears difficult to prevent this active MITM attack if the end-user machine is infected by malicious software. In fact, it seems all software-only defenses fail for such compromised end-machines. On the other hand, if the active MITM proxy is located remotely, as in DNS server pharming-based MITM attacks, consistency check techniques involving alternative communication paths could be used to detect if requests intended to be sent to one server actually terminate at another. This provides protection to TwoStep against active MITM proxies.

Comparison with Challenge Questions. *Challenge questions* [14] are now widely used for recovering or resetting forgotten passwords, as well as authenticating users. For example, Royal Bank of Canada allows users to register a primary computer with the bank for online banking by accepting a cookie on that machine. When a user signs in from this computer, she will be authenticated by her account number and password. When she signs in from a non-primary computer, authentication involves these, as well as a challenge question. ING Direct in Canada also uses challenge questions, but in contrast, prior to passwords. A user first provides her account number, then answers a challenge question. After a correct answer, a personalized site image is displayed, and the user is asked to enter her password. The user is supposed to enter her password only when seeing the site image, supposedly protecting the user password. One advantage of TwoStep over a combination of challenge questions and text passwords (both entered from keyboard) is that it is less vulnerable to naive keylogging attacks, because the graphical password part in the former is entered by mouse clicks.

4 Related Work

Graphical passwords can be largely classified into three categories: recognition-based, cued-recall, or recall-based. In recognition-based graphical passwords, users are required to recognize and then select a set of preselected images from a larger set. In cued-recall, the images cue the user, for example, to click a set of points on an image [3]. In recall-based, users are required to recall a password without any cues, such as drawing a doodle in Draw-A-Secret (DAS) [12]. We focus the remainder of our review here on recognition-based schemes. For a broader survey, see Chiasson [1, Chapter 2].

Deja Vu [6] is a recognition-based graphical password, which makes use of random art images, instead of photographs, to discourage users from selecting predictable

images. While randomly generated images can improve security, they also reduce usability. For example, it takes longer for users to remember random art images than photos, and less time to forget them. Passfaces [4] is another recognition-based scheme, using human faces as authentication images. A user's password consists of k faces, each of which must be chosen from a set of $n > 1$ faces in each round of the selection. While human faces are more memorable than text passwords, it was also found [5] that users usually choose predictable faces as their passwords, e.g., faces of their own race. In addition, female faces and "attractive" faces are chosen more often than male faces. Those biases make human faces less suitable as password components.

Story [5] is similar to Passfaces, but uses a variety of photos to form image portfolios, and encourages users to select photos to form a story to improve memorability. In Weinshall's scheme [15], a user is asked to answer a sequence of questions based on a shared set of images with the server. This scheme can resist shoulder-surfing attacks, but requires significant training and has usability issues, as well as security issues [11].

5 Concluding Remarks

TwoStep offers some advantages in countering common attacks against text passwords, such as naive keylogging and phishing. We have implemented [8] a variation of TwoStep (including the selection panel) as an optional mechanism for protecting on-line backup of user data in a Windows-based password manager, and it has been chosen and used on a regular basis by more than 4000 users, suggesting that a combination of text and graphical passwords is usable. An obvious and necessary next step is a user study, ideally both a lab study and a field study leveraging our real-world deployment.

Acknowledgements

Funding through an NSERC Discovery Grant, Canada Research Chair, and NSERC ISSNet is gratefully acknowledged.

References

1. Chiasson, S.: Usable Authentication and Click-Based Graphical Passwords. Ph.D thesis, Carleton University, Ottawa, Canada (January 2009)
2. Chiasson, S., Forget, A., Biddle, R., van Oorschot, P.C.: Influencing Users Towards Better Passwords: Persuasive Cued Click-Points. In: Proc. of HCI 2008 (September 2008)
3. Chiasson, S., van Oorschot, P.C., Biddle, R.: Graphical Password Authentication Using Cued Click Points. In: Biskup, J., López, J. (eds.) ESORICS 2007. LNCS, vol. 4734, pp. 359–374. Springer, Heidelberg (2007)
4. Real User Corporation. The Science Behind Passfaces (September 2001)
5. Davis, D., Monroe, F., Reiter, M.: On User Choice in Graphical Password Schemes. In: Proc. of 13th USENIX Security Symposium (August 2004)
6. Dhamija, R., Perrig, A.: Deja Vu: A User Study Using Images for Authentication. In: Proc. of 9th USENIX Security Symposium (August 2000)
7. Dhamija, R., Tygar, J., Hearst, M.: Why Phishing Works. In: Proc. of Human Factors in Computing Systems (April 2006)

8. 51Logon: Simplifying SignIn Experience (in Chinese), <http://www.51Logon.com>
9. Felton, E., Balfanz, D., Dean, D., Wallach, D.: Web Spoofing: An Internet Con Game. In: Proc. of the 20th National Information systems Security Conference (October 1997)
10. Florencio, D., Herley, C.: A Large-Scale Study of Web Password Habits. In: Proc. of the 2007 World Wide Web (2007)
11. Golle, P., Wagner, D.: Cryptanalysis of a Cognitive Authentication Schemes (Extended Abstract). In: Proc. of the 2007 IEEE Symposium on Security and Privacy (May 2007)
12. Jermyn, I., Mayer, A., Monroe, F., Reiter, M.K., Rubin, A.: The Design and Analysis of Graphical Passwords. In: Proc. of the 8th USENIX Security Symposium, August 23-26 (1999)
13. Kumar, M., Garfinkel, T., Boneh, D., Winograd, T.: Reducing Shoulder-surfing by Using Gaze-based Password Entry. In: Proc. of SOUPS 2007 (July 2007)
14. Rabkin, A.: Personal Knowledge Questions for Fallback Authentication. In: Proc. of the 2008 Symposium On Usable Privacy and Security (SOUPS), July 23-25 (2008)
15. Weinshall, D.: Cognitive Authentication Schemes Safe Against Spyware (Short Paper). In: Proc. of the 2006 IEEE Symposium on Security and Privacy (May 2006)

Design Principles for E-Government Architectures

Alain Sandoz

Université de Neuchâtel - CP 158 – CH-2009 - Switzerland

Abstract. The paper introduces a *holistic* approach for architecting systems which must sustain the entire e-government activity of a public authority. Four principles directly impact the architecture: *Legality*, *Responsibility*, *Transparency*, and *Symmetry* leading to coherent representations of the architecture for the client, the designer and the builder. The approach enables to deploy multipartite, distributed public services, including legal delegation of roles and outsourcing of non mandatory tasks through *PPP*.

Keywords: systems architecting, design principles, e-government platforms, multipartite distributed transactions.

1 Introduction

This paper's subject is *architecting a system capable of sustaining the entire e-government activity of a public authority*. The results are applied in Geneva's e-government program [1,2], for *ca.* 1 million residents and 100'000 local businesses, associations and *NGO*'s. Lower level authorities number *ca.* 100. Higher authorities exist at national and international levels. Each authority defines a jurisdiction under which lower-level authorities as well as residents, businesses and *NGO*'s live or operate. Foreign authorities can interact with our referential mid-level authority at the equivalent and at different levels (Geneva has many international organizations and shares its borders with France and the EU).

Each authority models, implements and operates a subset of regulatory aspects of society. Many interactions exist between such a referential and its environment. [3] lists *ca.* 800 official procedures for the State of Geneva alone. Organizational channels, rules and procedures change regularly, whereas traditional technical channels (teller, postal mail, telephone, fax) were only recently extended with ICT's and the internet. Integrating this new *channel* into government interactions is called e-government and requires resources and components that were not used for the purpose of interacting with society before. To identify, implement and integrate these components in order to sustain *parts of* or *the whole of* e-government activities is the task of building an e-government platform (noted *EGP*).

The "parts of" and the "the whole of" approach are fundamentally different. The former leads to incremental strategies for e-government. Projects "start small" with easy to deliver services, or emphasize communication with citizens and develop a portal, etc. Over time, many incremental projects encounter difficulties.

The latter considers an *EGP* from the start as a *system*. Architecting complex systems is delicate because of the global impact of *each* component and of the importance

of component *interactions* [4]. An earlier success with Geneva's system for e-voting through internet [5] led us to develop this approach on the larger scale of e-government. We designed a small set of principles for architecting an *EGP*. The principles are *Legality*, *Responsibility*, *Transparency* and *Symmetry*. Their formulation speaks to policy and decision makers confronted with problems in e-government like complexity, scale and costs; expectations of citizens, choice, added value, and trust; dissemination, impact of technology, and change in society.

The following section presents the principles and their impact on the architecture. Section 3 briefly examines the consequences at the systems level. This paper is a short version of [6] which develops the approach and illustrates it with examples and technical issues related to architecting an *EGP* in this manner.

2 Principles for Architecting an *EGP*

Architecting a system ("architecture" comes from the Greek words for "principle" and "construction") consists in designing abstractions and representations of structural characteristics which enable to understand, build, repair, and reproduce the system all through its life cycle. Modern systems architecting [4] sees the architect principally as "an agent of the client, not the builder". The discipline relies on six foundations: 1) *a systems approach*; 2) *a purpose orientation*; 3) *a modeling methodology*; 4) *ultraquality implementation*; 5) *certification*; and 6) *insights and heuristics*. Because governments are builders of their own information infrastructure, they tend to build e-government infrastructures as an appendix to existing IT platforms. These are often juxtapositions of legacy applications, loosely connected through an intranet for "interoperability" (for this reason one occasionally encounters more architects than programmers in an e-government program). Initiatives, like the US DII COE [7] define *Enterprise Architecture Frameworks* [8] and standardization processes [9] for government information systems (see [10] on IS and enterprise architectures). Though uncommon, it is necessary to approach the design of an *EGP* from the client's side, because the client, *i.e.* society as a whole, *is not an enterprise*. So even if e-government has a track record of more than 10 years, building an *EGP* can be *complex* and *new* in many respects. Designing this new system *vs.* adding a feature to the old one is comparable to "designing systems safety" *vs.* "reliability engineering" in aeronautics [11].

Legality principle (LP). In e-government a *user* is an actor (a citizen, a non-resident, a business, a *NGO*, an association, another administration, a machine, or any identified or unidentified entity) that interacts with an administration through the internet. *We do not assume* that a contract between the administration and the user exists. The first principle aims at protecting the user's legal and civil rights, if any, in the jurisdictions under which the administration operates. **LP:** Any operation suggested to a user on the *EGP* and all the consequences of its execution *by the user* must be legal and respect the user's legal and civil rights within the jurisdictions under which the authority operates.

In the absence of e-government, interfaces between user and administration go through human operators who are instructed to respect users' rights *at the interface*. Operations requested from a user are of the type "make a declaration", "sign a

document”, “deliver information”. All downstream operations are executed in the authority’s domain of responsibility, either manually or semi-automatically. Traceability is enforced *ad hoc* using *files* of paper or electronic records for each *business case*.

Under *LP*, a user cannot be lead to execute an interactive operation on the *EGP* if the underlying application is not *certified*. *Ultraquality implementation* implies that if there is any chance that *LP* will not be respected because of a loophole in the design, then the design must be scrapped. Therefore, in practice, it is impossible to authorize *interactive* access by users to legacy applications on the *EGP*. Interactive operations must be simple and traceable (*e.g.* upload a file, trigger a transition in a workflow) with *certified* effects. At any time, users must be able to reconstruct the trace of operations they executed on the *EGP*. For these reasons, interactive operations are deliberately confined to the *EGP* (see Figure 1) and the resulting operations on the legacy infrastructure of the administration are executed *asynchronously*. This implies that the stovepipe model of an administration’s information system infrastructure need not be extended to the *EGP*.

Responsibility principle (RP). “Who is the user?” is a delicate question, often oversimplified by security technicians who treat the problem at the identification and authentication level of *enterprise directories*, using logins of physical persons as a basis for profile and role management. In e-government this is not possible because *society is not an enterprise* and e-government cannot impose a contract on every citizen. Managing logins and roles for all people interacting with an administration through the web in the name of their employers, of their customers, of the *NGO*’s they are active in, and in their own, is meaningless, expensive, and insecure for an administration (and moreover violates *LP*).

RP: Each *operation* executed on the *EGP* must be attributed to a unique identified legal personality. This person is responsible for the execution of the considered operation and for all its publicized and certified consequences.

The legal personality can be a physical person, a moral person (company, *NGO*, etc.), or the government itself (which is a legal personality in its own jurisdiction). If the user is untraceable, the responsible personality must be the state. *RP* sets the basis to solve the *delegation problem*, *e.g.* an agent or employee that operates in the name of a customer or employer is responsible legally towards the latter according to the contract binding these two entities, whereas the customer or the employer is responsible towards the state for matters concerned with the given administrative procedure, *i.e.* for all the operations executed by the agent.

For each operation executed on the *EGP*, it is necessary to identify the legal personality assuming responsibility. From a design point of view, this draws lines between 1) operations executed under legal responsibility of the state (*e.g.* decision steps in e-administrative processes, notifications, and everything that runs on legacy systems), 2) operations executed under the *control of the administration* but under the *responsibility of the user* (*e.g.* upload of a form), and 3) operations executed out of the state’s scope of responsibility, *i.e.* operations executed on infrastructures out of the administration’s control (user PC’s, company mainframes. Operations of the second type, in particular, must be to be certified.

Transparency principle (TP). In the absence of e-government, government employees use extensive knowledge to answer users' requests: from terminology, regulation, organizational aspects, along with the knowledge of stakeholders and of the legacy applications used during the procedure, an employee might need years of training to work efficiently. Take the human actor out of this process and the missing knowledge becomes a problem. Shifting the *necessity to know* to the user is disloyal and impossible in practice. Consequently, the *EGP* must be architected so that this knowledge becomes unnecessary. **TP.** Any organizational characteristic of a concerned party which is not explicitly necessary to perform an operation on the *EGP* must not be reflected in that operation.

In particular, any function used by more than one agency (*e.g.* login, payment, trace, geo-localization, directory, support, delegation, etc.) must be seen by users as a unique service instance. Such services are *transversal* and be implemented in the *EGP*. Paradoxically, this establishes the *EGP* as the central locus for *dematerializing* administrative processes *inside* the administration.

Symmetry principle (SP). The last principle is the least intuitive. **SP:** If a function is necessary for an *EGP* to operate correctly, but is not directly linked to the state's sovereignty, it should be architected on the *EGP*, if at all, in such a way that it can be provided externally.

Consider e-forms. Internet transactions require that users transmit formatted texts. Syntax and low level semantics of inputs are verified online. Etc. This service is necessary for any *EGP* to operate. *If* the service is provided by the *EGP*, it should be unique (by *TP*) and furnish every form the administration uses. However, enabling users to *complete forms* online is not an issue related to sovereignty of the state. Accordingly the e-forms service should be implemented on the *EGP*, if at all, so that an external service provider can do the *same*. This comprises technical aspects like transparent invocation of the external service from the *EGP* and data transfer to the *EGP*, as well as functional aspects like publishing verification rules and data or file structures. So a service which is built according to *SP* can be *shared among administrations without loss of sovereignty on behalf of the related authorities*. This enables administrations to overcome the small-scale syndrome, a killer factor for local governments seeking a presence on the web.

3 Design of the EGP

An *EGP* lies on a foundation of legacy systems which compose an administration's IT infrastructure. In the absence of e-government, this core is isolated from the internet. It is organized in stovepipes for legal or for historical reasons. Due to pressure on costs, some functions are shared among agencies, possibly leading to an enterprise architecture. Interoperability [12] of sub-systems is a goal, but is not a requirement (according to *TP*, the authority can organize as it wishes, as long as specifics are not reflected to external users). Operations are executed on the infrastructure under the legal responsibility of the authority.

External users also possess an infrastructure operated under their sole legal responsibility. *LP* does not apply to either of these two domains because the principle is

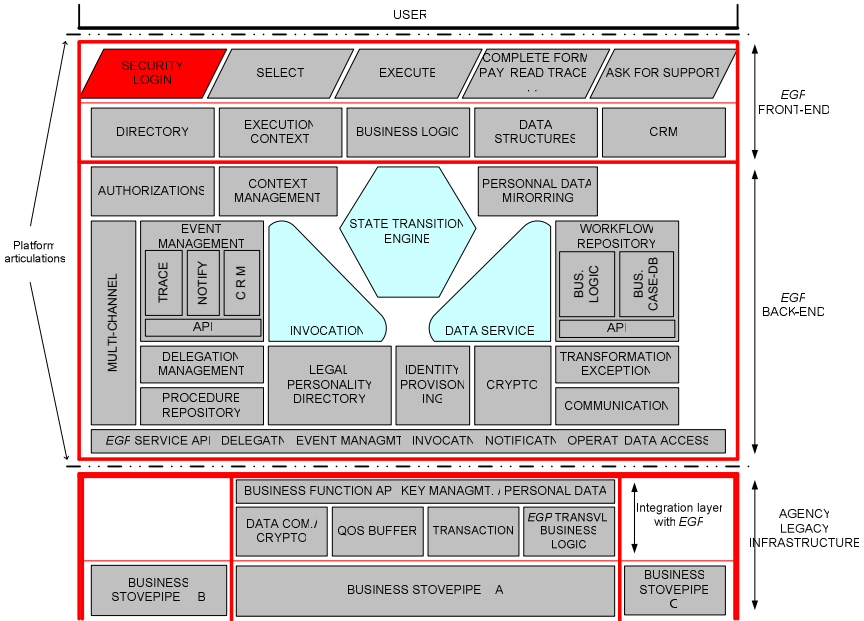


Fig. 1. Articulation and organization of the EGP

only concerned with operations exposed on the EGP interface, and this applies *a priori* to neither one. This is represented by articulations in Figure 1. The articulations determine the mechanisms which will be used to control interactions between the EGP and both the internet and the legacy infrastructure. All mechanisms must be certified according to all four architecting principles.

When discussing *TP*, we noted that users must be given simple functions to invoke, like “Select”, “Execute”, “Complete form”, “Pay”, “Ask for help”, “Read a trace” or “Delegate an operation”. The nature of interactions on the EGP is *multipartite, distributed and asynchronous*. At any moment, all actors should see consistent *views* of a given business case state. The *execution context* of a user contains different kinds of elements like workflow instances, documents, partially or fully completed forms, attributes, messages, invoices, etc. This depends on regulatory aspects and to some extent is independent of the administration’s organization. The application that manages the execution context of a user is designed according to *TP*. It contains a state transition engine, and mechanisms to control access to resources according to dynamic rules.

4 Conclusion

Implementing the four principles, or any others, in the EGP’s architecture is a choice. Platforms that don’t respect *LP, RP, TP* or *SP* exist, operate and are correct. However, this set of design principles applies to representations [10] that pertain to the client,

the designer and the builder of an *EGP*. The principles enable systematic certification of conformity and global coherency in the operation and continuous development of a complex e-government system.

The principles are declined into structural constraints of *EGP*'s, like articulations or predicates on the locality of data persistency. Complex transactions (*i.e.* multipartite, distributed and asynchronous) can be implemented, respecting all four principles and enabling in depth reengineering of business processes without requiring in breadth remoulding of the enterprise architecture.

It is possible for an administration to provide services to others (*i.e.* hosting services for other authorities) and to subscribe to external services (public, or private through *PPP* [13]) without loosing sovereignty. This enables to position an *EGP* as a member of a community which can contribute to deliver exhaustive shared e-government services on a large scale.

References

1. Secrétariat du Grand conseil: Projet de loi ouvrant un crédit d'investissement de 30 850 000 F pour le développement de l'administration en ligne. Geneva (2008), <http://www.ge.ch/grandconseil/data/texte/PL10177.pdf>
2. Sandoz, A., Haenni, N., Eudes, J.-R.: Addressing and Protecting Distributed Resources. In: e-Government Architectures using Multiple Digital Identities, ECEG (2005)
3. e-Government Standards: eCH0070 Inventory of Public Services, Ver. 2.5 (July 2007), <http://www.ech.ch>
4. Maier, M., Rechtin, E.: The Art of Systems Architecting, 2nd edn. CRC Press, Boca Raton (2000)
5. Chevallier, M., Warynski, M., Sandoz, A.: Success Factors of Geneva's e-Voting System. Electronic Journal on e-Government 4.2 (2006), <http://www.ejep.com>
6. Sandoz, A.: Design Principles for E-Government Architectures, RT-0901-ASA, University of Neuchâtel (January 2009)
7. Frazier, G.: The DII COE: An Enterprise Framework, Journal of Defense Software Engineering (October 2001), <http://www.stsc.hill.af.mil/CrossTalk/2001/10/frazier.html>
8. CIO Council: Federal Enterprise Architecture Framework (September 1999), <http://www.cio.gov/Documents/fedarch1.pdf>
9. NIH Enterprise Architecture Standards Development Process, <http://enterprisearchitecture.nih.gov/>
10. Zachmann, J.A.: A Framework for Information Systems Architecture. IBM Systems Journal 26.3 (1987)
11. Leveson, N.G.: Safety in Integrated Systems Health Engineering and Management, draft paper for the NASA AISHEM Forum (November 2005)
12. Gov. of the Hong Kong SAR: The HKSARG Interoperability Framework (December 2007)
13. Sandoz, A., Eudes, J.-R., Prévot, R.: Public-Private Partnership. In: e-Government: a Case Implementation, MCETECH, Montreal (January 2008)

A Proposed Intelligent Policy-Based Interface for a Mobile eHealth Environment

Amir Tavasoli and Norm Archer

McMaster University
1280 Main Street West,
Hamilton, Ontario, Canada L8S 4M4

Abstract. Users of mobile eHealth systems are often novices, and the learning process for them may be very time consuming. In order for systems to be attractive to potential adopters, it is important that the interface should be very convenient and easy to learn. However, the community of potential users of a mobile eHealth system may be quite varied in their requirements, so the system must be able to adapt easily to suit user preferences. One way to accomplish this is to have the interface driven by intelligent policies. These policies can be refined gradually, using inputs from potential users, through intelligent agents. This paper develops a framework for policy refinement for eHealth mobile interfaces, based on dynamic learning from user interactions.

Keywords: Mobile eHealth, adaptive interfaces, intelligent agents, policy-based networking.

1 Introduction

Mobile wireless systems continue to increase in popularity in many disciplines and applications, for both leisure and work. These range from wireless online games to teleconferencing and mobile commerce applications such as ticketing and e-mail [1]. But there has also been a great deal of interest in mobile wireless applications in supporting healthcare [2]. As such applications become more sophisticated, there is an inevitable trend towards the direct support of patients. To accomplish this individual support for the range of users who are either healthcare practitioners or patients managing their own health, an intelligent environment would be extremely useful to help the system adapt to users with a wide range of needs. It is also expected that mobile eHealth applications will maintain or improve patient quality of life while at the same time reducing system costs, paper forms, delays and errors [3].

In recent years much work has been done in order to solve technical issues around mobile wireless networks. Since most people that work with these systems are novices, the system should be simple and convenient enough for anyone to use - i.e. the system should be usable and adaptable [3]. The user interface (UI) clearly plays a key role in these features, and this is especially important in mobile systems where the device screen is limited in size [4, 5].

Advances in policy-based networking (PBN), coupled with AI, have provided tools that will not only manage applications effectively, but will support adaptability to

different users who may display a wide range on needs. This is particularly true with mobile eHealth applications, where some users such as nurses and physicians may be very familiar with healthcare requirements, but novices at using mobile devices. On the other hand, others such as patients may be novices at managing their own healthcare and more or less familiar with the use of mobile devices. To cater to this broad range of end users requires a system interface that is not only highly usable but adaptable to a wide range of users. Actually an interface that can fully address these issues can be used not only in eHealth but in other areas like e-commerce [6].

Intelligent User Interfaces (IUI), which apply AI concepts to UIs, have been used to handle standard application requirements such as helping end-users to manage complex systems [7, 8]. IUIs are complex, and researchers have studied how to implement them efficiently for some time. For example, Langley [9] suggested using machine learning for implementing IUIs. Various frameworks have also been proposed for managing IUIs for mobile systems [8, 10].

In this paper, we propose a framework for a policy-based interface that can adapt itself to user needs with the help of AI methodology, using a policy refinement method that refines the policy gradually, according to both user choices and learning from user actions. This implements the IUI concept in mobile eHealth applications, through a novel combination of AI and policy-based networking.

2 Policy-Based Networking

PBN [11] is a way of managing network resources using pre-defined business policies. It is actually an efficient automated extension of manual classical management methods, allowing administrators to effectively manage resources according to their business needs. Each policy is a set of rules and instructions that dictates how the network is to use its resources [11]. PBN mostly deals with issues around network management. However, in this work we focus on the use of policies that can be used to build and manage UIs, a key business aspect of the eHealth environment because of the wide range of user capabilities and requirements. This becomes an even more critical issue because of limitations due to the small screen size of mobile devices [4].

3 Agents in Policy-Based Environments

Agents are an innovative concept that can help to automate network management in policy-based mobile environments [12]. In this paper we extend PBN with a special kind of policy which is quite different from traditional policies. Traditional policies are built from rules [11], which are conditional if-then-else statements. Implementing these rules in specific situations requires matching situation parameters with policy conditions. If there is a successful match, the condition body is applied. We will refer to these policies as 'solid policies'.

This paper implements a special kind of policy which is actually a learner agent. In our system, the agent receives user inputs and its resulting actions are used to build or modify an intelligent interface. The agent's purpose is to derive an output, which will lead to the best interface that it can provide, when it receives user input. What we

really need is an intelligent policy. We call these policies ‘dynamic policies’. To implement intelligent agents in these environments there are various artificial intelligence (AI) techniques available, including artificial neural networks (ANNs), fuzzy logic, etc. For the purpose of building and modifying dynamic policies we have chosen ANNs due to their powerful generalization, speed, flexibility and adaptability [13].

The properties of ANNs will, first, allow the interface agent to adapt to user needs automatically and dynamically. Second, the agent will be able to generalize a user’s screen with some properties and then adjust to the best screen for users with somewhat different needs. Note that ANNs have been used in some cases to develop conversational interfaces for mobile users [8, 14]. The next section discusses the translation of user choices into input vectors, how to interpret the output to build the required interface, and how to use the ANN to support policies.

4 Policy-Based Interfaces

A policy-based interface framework includes both profiles and policies. Each user device or machine will have a user profile and an interface profile. There will be a Local Interface Agent (LIA) at each user machine. On the server side we will have a Central Interface Agent (CIA). Although it is preferable to maintain agents locally (on the devices) because of speed and network bandwidth savings, handheld devices may not be powerful enough to support LIAs. In our proposal, the Local Interface Policy (LIP) does not require a great deal of local computation due to its simple design. Nevertheless, it would still be possible if desired to maintain all of the agents on the server. Since network and device bandwidth may not be high enough to handle this situation, the choice of where to maintain the agents requires a performance test to determine the optimal configuration.

For the purpose of this paper, we will assume that the network and the handhelds will have the bandwidth and computing capacity to run the agents locally.

4.1 User Profiles

In this environment each user will have a profile which contains user information such as diseases that afflict the user, drugs being taken, health status (heart rate, blood sugar level, etc.) For applying these profiles to the agents, the attributes will need to be transformed into a number vector. Data [15] in these profiles are categorical data, so a codebook would be constructed from a set of numerical codes assigned to each item, and stored in the codebook. More complex conversion methods could be used if required, based on existing standards for medical data.

4.2 Local Interface Policy

There are two types of agents in the framework. One, the LIA, controls the Local Interface Policy (LIP) and the one Central Interface Agent (CIA) controls the Central Interface Policy (CIP). The LIA has a dynamic policy that builds and maintains the LIP. To build the interface, user profile information will be sent to the LIP agent as inputs, and the agent outputs are used to set up links and other relevant information

such as page configurations in the local device. Each time the local system boots up, the interface will be updated according to the policy that relates to that system.

It is worth noting that what is meant by an interface component is anything appearing on the screen, ranging from hyperlinks to icons. The technique that will be used for building the interface is as follows. First the user profile is converted to a number vector, which becomes the input to the LIA. The output of the LIA will be a number¹ vector that has been generated intelligently to build the interface. For example it sets priorities for the arrangements of the interface components. Literally these are the priority levels of the components². Other properties such as size and colour can also be taken into account. It is expected that issues such as component placement, size, and colour would be significant considerations in designing and maintaining optimal interfaces for individual users.

4.3 Interface Profile

Each user will have an Interface Profile, which records user changes to interface components. This ensures that user choices will be preserved for future sessions and will not be revised by the dynamic policy each time a new session is initiated., and they will be saved for future sessions with the device since they represent user choices and preferences.

4.4 Central Interface Policy

The CIP is a server-based dynamic policy that keeps track of the interface choices of all the users dynamically. This will help in designing a suitable interface for all new users. From the agent perspective, the inputs to this agent are profiles of all of the current users of the system and their preferences in building their interfaces. Each time [11] an interface is created for a new user, the initial LIP will be inherited from the CIP, which the user can then be modified and adapted to suit that particular user's needs. Each new LIA that is built from the CIA is an object that has inherited the characteristics of the CIA. This object thus includes all of the current knowledge about interfaces in the network. As more users enter the system, the CIA will learn more about the range of inputs needed, and what this translates into in terms of outputs.

We need to consider the fact that just copying the ANN weights to an LIA most probably will not result in the agent inheriting the properties desired for that specific UI, since the CIA has learned its properties from a large number of users. It might therefore take too long for the LIA to adapt to that particular user's choices. Other methods for transferring learned properties that can be considered include training the LIA with the inputs and outputs of the CIA a controlled number of times.

¹ Input numbers are integer numbers and outputs can be decimal numbers. Since outputs are used for prioritizing, output numbers only need to be comparable.

² Each component in the interface is related to an item in the user profile. For example for disease we have a component to show diseases. Because of the nature of feed-forward back-propagation neural networks the number of inputs and outputs can differ, and any relation needed can be assumed. The neural network will choose the preference pattern best suited to this particular user.

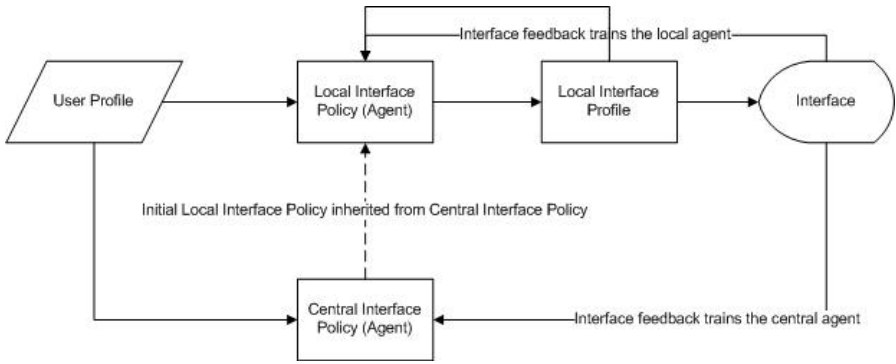


Fig. 1. Logical Relationships of Network Profiles, Policies, and Agents

4.5 Learning

When a new user is added to the system the user’s interface will inherit its initial LIP from the CIP. The interface generated will be a representation of all the previous system interfaces for current users. The new user will modify the interface to suit his or her tastes, by dragging components to different locations, changing a link hierarchy in a menu tree, or changing colours or fonts of a component. The user may otherwise begin working with the page as it currently exists. In any case, user hits on specific components on each page will be tracked, to enable training of the local LIAs and the CIA with the observed activity (that is, the interface feedback to LIPs and CIP will train and adapt them to user choices). If the user changes the place of a specific component, the priority of that component may change. If this continues regularly, then the priority of that component will increase relative to the components that are being used less. In this way, interfaces will gradually move closer to user needs and the corresponding agents will learn more about user preferences. LIAs will only be trained for their specific users, but the CIA will receive interface feedback from all the users, based on the structure that we have just described.

The LIP will save user choices about such decisions as the placement of components on the screen, and the results of the LIA will be ignored when it comes to this profile. This is because, when the user changes component priorities, it takes time for the agent to learn, but the user expects to see an immediate change. The Interface Profile will save user choices about properties of components such as place and size, and this profile will be taken into account each time the LIA is trained. This places an emphasis on the concrete choices of users. These concrete choices need to be emphasized locally, but emphasizing them globally through the CIA may overwhelm the CIA’s learning process. Therefore the User Profile will only be transmitted to the CIA each time it is changed through Interface feedback. The learning process is demonstrated in the logical framework shown in Figure 1.

5 Future Work

The initial priority for this project will be to design agents that will provide acceptable performance in the context of eHealth mobile environments, using published narratives

about developing high quality ANNs [13, 16]. The second major project phase will be to design and implement a test interface, using simulations and human participants to establish the effectiveness of the system before it is actually considered for production applications. Additionally this work could be extended in a whole PBN framework, including the possibility that dynamic policies can be used to manage all of the resources in the network and not just the interface.

References

- [1] Buellingen, F., Woerter, M.: Development perspectives, firm strategies, and applications in mobile commerce. *Journal of Business Research* 57, 1402–1408 (2004)
- [2] Orwat, C., Graefe, A., Faulwasser, T.: Towards pervasive computing in health care: A literature review. *BMC Medical Informatics and Decision Making* 8 (2008)
- [3] Archer, N.: Mobile eHealth: Making the case. In: Kushchu, I. (ed.) *Mobile Government: An Emerging Direction in e-Government*, pp. 155–170. Idea Group, Hershey (2007)
- [4] Tarasewich, P.: Wireless devices for mobile commerce: User interface design and usability. In: Mennecke, B., Strader, T. (eds.) *Mobile Commerce: Technology, Theory, and Applications*, pp. 26–50. Idea Group Publishing, Hershey (2003)
- [5] Zhu, W., Nah, F.F.-H., Zhao, F.: Factors influencing user adoption of mobile computing. In: Mariga, J. (ed.) *Managing E-Commerce and Mobile Computing Technologies*, pp. 260–271. IRM Press, Hershey (2003)
- [6] Liu, S.-P., Tucker, D., Koh, C.E., Kappelman, L.: Standard user interface in e-commerce sites. *Industrial Management & Data Systems* 103, 600–610 (2003)
- [7] Höök, K.: Steps to take before intelligent user interfaces become real. *Interacting With Computers* 12, 409–426 (2000)
- [8] O’Grady, M.J., O’Hare, G.M.P.: Intelligent user interfaces for mobile computing. In: Lumsden, J. (ed.) *Handbook of Research on User Interface Design and Evaluation for Mobile Technology*, vol. 20. IGI Global, Hershey (2008)
- [9] Langley, P.: *Machine learning for adaptive user interfaces*. LNCS, vol. 1303, pp. 53–62. Springer, London (1997)
- [10] Mitrovic, N., Royo, J.A., Men, E.: Adaptive user interfaces based on mobile agents: Monitoring the behavior of users in a wireless environment. In: *Symposium on Ubiquitous Computation and Ambient Intelligence*, Madrid, Spain (2005)
- [11] Kosiur, D.: *Understanding Policy-Based Networking*. Wiley, New York (2001)
- [12] Ganna, M., Horlait, E.: On using policies for managing service provisioning in agent-based heterogeneous environments for mobile users. In: *Sixth IEEE International Workshop on Policies for Distributed Systems and Networks*, Stockholm, Sweden (2005)
- [13] Bailey, D., Thompson, S.: How to develop neural networks. *AI Expert* 5, 38–47 (1990)
- [14] Toney, D., Feinberg, D., Richmond, K.: Acoustic features for profiling mobile users of conversational interfaces. In: Dunlop, M.D. (ed.) *Mobile HCI 2004*. LNCS, vol. 3160, pp. 394–398. Springer, Heidelberg (2004)
- [15] Weiss, S., Indurkha, N., Zhang, T., Damerau, F.: *Text Mining: Predictive Methods for Analyzing Unstructured Information*. Springer, Heidelberg (2004)
- [16] Tsoukalas, L.H., Uhrig, R.E.: *Fuzzy and Neural Approaches in Engineering*. Wiley, New York (1997)

Verification of Information Flow in Agent-Based Systems

Khair Eddin Sabri*, Ridha Khedri**, and Jason Jaskolka***

Department of Computing and Software
Faculty of Engineering
McMaster University
{sabrike,khedri,jaskolj}@mcmaster.ca

Abstract. Analyzing information flow is beneficial for ensuring the satisfiability of security policies during the exchange of information between the agents of a system. In the literature, models such as Bell-LaPadula model and the Chinese Wall model are proposed to capture and govern the exchange of information among agents. Also, we find several verification techniques for analyzing information flow within programs or multi-agent systems. However, these models and techniques assume the atomicity of the exchanged information, which means that the information cannot be decomposed or combined with other pieces of information. Also, the policies of their models prohibit any transfer of information from a high level agent to a low level agent. In this paper, we propose a technique that relaxes these assumptions. Indeed, the proposed technique allows classifying information into frames and articulating finer granularity policies that involve information, its elements, or its frames. Also, it allows for information manipulation through several operations such as focusing and combining information. Relaxing the atomicity of information assumption permits an analysis that takes into account the ability of an agent to link elements of information in order to evolve its knowledge.

The technique uses global calculus to specify the communication between agents, information algebra to represent agent knowledge, and an amended version of Hoare logic to verify the satisfiability of policies.

Keywords: Global calculus, Information Algebra, Agent Knowledge, Information Flow, Hoare Logic.

1 Introduction and Motivation

Security is an important aspect that ought to be considered during the software development life cycle. An early detection of a system vulnerability would reduce

* This research is supported by the University of Jordan.

** This research is supported by Natural Sciences and Engineering Research Council of Canada (NSERC).

*** This research is supported by NSERC USRA (Undergraduate Summer Research Assistantship).

the cost of addressing it. Information security has three major facets: confidentiality, integrity, and availability. The confidentiality of information ensures that only those with sufficient privileges may access a pre-specified set of information. Access control mechanisms are used to protect information from being read or modified by unauthorized agents. Cryptography provides confidentiality in open environments. Once the information is released, it can be transmitted by mistake or malice to unauthorized users. Analysis techniques and prevention and detection mechanisms are necessary to track and control information flows within a system to prevent information from leaking to unauthorized agents.

Models are proposed to anticipate the authorized paths that information should follow and articulate rules for its circulation. For instance, Bell-LaPadula model [2] has its origin in the military and it is widely used in many organizations. The model gives security labels to objects (e.g., information) and subjects (e.g., agents). Each object is considered as one component that cannot be decomposed and is assigned one security level. The security levels form a lattice in such a way that the highest element of the lattice is the most sensitive one. Bell-LaPadula model describes a set of rules that proscribe any flow of information from a high level to a lower one. We point to two issues on the Bell-LaPadula model. First, the model prohibits any transfer of information from a high level agent to a low level agent. Second, Bell-LaPadula model does not take into consideration policies on composite objects. For example, an agent should be allowed to access separately a list of name and lists of drugs administered by a hospital but should not be able to link a patient to a drug.

The Chinese Wall [3] is another model where access to information is not constrained to its security level. Instead, datasets are grouped into conflict of interest classes and an agent can have an access to an information of one of these datasets. For example, assume that there are two banking systems A and B and an oil company C . One policy can group A and B into a conflict of interest class so that an agent can have an access either to A or to B . At the same time, an agent can have an access to C . However, one may want to state a policy that an agent should not be able to associate information together from the bank system and the oil company, such as the investment of an employee in a company. Similar policies cannot be articulated within the Chinese Wall model.

The manual verification of information flow is extremely demanding for both time and resources especially in complex systems. Therefore, formal methods that can constitute the background for sound automation of the analysis of information flow policies become necessary. For example, Security Process Algebra (SPA) [6] is a CCS-like process algebra where actions are partitioned into two security levels (*high* and *low*). Using the notion of bisimulation, SPA is used to verify that no information flow is possible from high-level user to low-level user. SPA can specify concurrent systems and detect direct and indirect information transmission but it deals only with two security levels and message passing is not specified. Varadharajan [14] proposes an extended Petri-net formalism to model information flow security requirements such as security classes of the output cannot be lower than the security class of any of the received messages. Alghathbar

et al. [1] use a logical-based system called FlowUML to validate information flow policies of UML sequence diagrams. Their aim is to detect security flaws at an early stage of the software development life cycle. Typing systems [8,9,15] have been widely used for analyzing information flow within the code of programs but not at an earlier stage. Analyzing composite-information flow is not addressed in the techniques presented above.

In this paper, we propose a technique for verifying information flow in agent-based systems. The technique allows for classifying information into frames (i.e., classes) and articulating finer granularity policies that involve information, its elements, or its frames. Also, the technique allows for analyzing policies governing the flow of composite-information formed from pieces of information of different attributes. For example, a student's name can be seen as an atomic information as it contains only one of the student attributes. Knowing such information may not violate a security policy. However, an unauthorized knowledge of a composite information that consists of a student's name and her grades could cause a security breach in the registrar system. Also, the technique removes the restriction that pieces of information should have the same classification in the agents' knowledge.

In Section 2, we introduce the required background. In Section 3, we use an example to illustrate the proposed technique. In Section 4, we present the proposed technique. In Section 5, we bring in related work with a discussion. We conclude in Section 6.

2 Background

The proposed technique is used for analyzing the composite-information flow policies in a distributed system where agents are communicating by sending messages. To specify the information flow, the developer needs to specify the knowledge of each of its agents and their communications. We use information algebra [10] for knowledge modeling and global calculus [4] for capturing the communication among agents. The verification is based on an amended version of Hoare Logic [7]. In the following subsections, we introduce these formal settings.

2.1 Information Algebra

In [10], Kholas and Stark explore connections between different representations of information. They introduce a mathematical structure called *information algebra*. This mathematical structure involves a set of information Φ and a lattice D .

In the following definition and beyond, let (D, γ, λ) be a lattice and x and y be elements of D called frames (also called domain in [10]). Let \preceq be a binary relation between frames such that $x \gamma y = y \Leftrightarrow x \preceq y$. Let Φ be a set of information and φ, ψ, χ be elements of Φ . We denote the frame of information $\varphi \in \Phi$ by $d(\varphi)$. Let e_x be the empty information over the frame $x \in D$, the operation \downarrow be a partial mapping $\Phi \times D \rightarrow \Phi$ to restrict an information to a specific domain, and \cdot be a binary operator to combine pieces of information. For simplicity, to denote $\varphi \cdot \psi$, we write $\varphi\psi$.

Definition 1 (Information Algebra [10]). An information algebra is a system (Φ, D) that satisfies the following axioms:

1. $(\varphi\psi)\chi = \varphi(\psi\chi)$
2. $\varphi\psi = \psi\varphi$
3. $d(\varphi\psi) = d(\varphi) \vee d(\psi)$
4. $x \preceq y \Rightarrow (e_y)^{\downarrow x} = e_x$
5. $d(\varphi) = x \Rightarrow \varphi e_x = \varphi$
6. $\forall(x \mid x \in D : d(e_x) = x)$
7. $x \preceq d(\varphi) \Rightarrow d(\varphi^{\downarrow x}) = x$
8. $x \preceq y \preceq d(\varphi) \Rightarrow (\varphi^{\downarrow y})^{\downarrow x} = \varphi^{\downarrow x}$
9. $d(\varphi) = x \wedge d(\psi) = y \Rightarrow (\varphi\psi)^{\downarrow x} = \varphi(\psi^{\downarrow x \wedge y})$
10. $x \preceq d(\varphi) \Rightarrow \varphi\varphi^{\downarrow x} = \varphi$

□

As we will illustrate in Section 4, having an information algebra to represent agent knowledge allows verifying policies that involve a flow of composite information as well as expressing policies similar to that of Bell-LaPadula and Chinese Wall models as shown in Section 5.

2.2 Global Calculus

In [4], Carbone et al. introduce a typed calculus to specify communication-centred systems called global calculus. We use global calculus to specify the communication between agents as it gives the view of messages moving between agents. This is the global view of the system to be analyzed. In contrast, π , endpoint, CCS, and CSP calculi give the view of the local behaviour of each agent. Also, global calculus is expressive as it can be used to represent initiating a session, in-session communication, branching, conditional, and recursion. Below, we give a part of the global calculus syntax, taken from [4], that we use in this paper.

$I ::= A \rightarrow B:ch(\tilde{v}\tilde{s}).I$	(initiation of a session)
$\mid A \rightarrow B:s(op, e, y).I$	(communication over a session channel s)
$\mid x@A=e.I$	(assigning the value of e to x at A)
$\mid \text{if } e@A \text{ then } I_1 \text{ else } I_2$	(condition)
$\mid 0$	(inaction)

The terms I_1 and I_2 are called interactions, ch is a service channel, s is a session channel, \tilde{s} is a vector of session channels, A and B are agents, and x and y are local variables in each agent.

2.3 Hoare Logic

Hoare logic is used to reason program correctness with respect to program specifications rather than in terms of how the program could be executed. Verification is based on the Hoare triple $\{P\} S \{Q\}$. In the triple, S is a program statement or a sequence of statements. The predicate P is the precondition that characterizes the initial states for which the program is being defined. The predicate Q is the postcondition that specifies the final states after the execution of the program. In order to verify $\{P\} S \{Q\}$, one needs to prove that $P \Rightarrow wp(S, Q)$

$$\begin{array}{l}
\text{(EMPTY)} \frac{}{\{P\} \text{skip} \{P\}} \quad \text{(ASSIGN)} \frac{}{\{P[x/E]\} x:=E \{P\}} \quad \text{(COMP)} \frac{\{P\} S \{Q\}, \{Q\} T \{R\}}{\{P\} S;T \{R\}} \\
\text{(COND)} \frac{\{B \wedge P\} S \{Q\}, \{\neg B \wedge P\} T \{Q\}}{\{P\} \text{if } B \text{ then } S \text{ else } T \{Q\}} \quad \text{(LOOP)} \frac{\{B \wedge P\} S \{P\}}{\{P\} \text{while } B \text{ do } S \{ \neg B \wedge P \}}
\end{array}$$

Fig. 1. Hoare logic inference rules

where $wp(S, Q)$ is the weakest precondition for the program S and the postcondition Q . The weakest precondition can be found by using the inference rules of Hoare logic given in Figure 1.

To use Hoare logic in verifying information flow, we rephrase the weakest precondition inference rules such that they can be applied to distributed systems specified using global calculus and information algebra.

3 Illustrative Example

We give an example to illustrate the proposed technique in analyzing the composite-information flow in a distributed system. We will use this example as a running example throughout this paper. Our illustrative system consists of four agents: Coordinator Agent (CA), Operations Officer (OO), Analyst Agent (AA), and Public Relation Agent (PR). These agents are communicating by sending messages that contain information. A message also specifies the frames to be associated with the transmitted information in the receiver's knowledge.

The CA sends a *mission* to the OO who collects data regarding this mission and sends it to the CA. The *data* communicated to the coordinator can be classified as either *country*, *company*, or *employee*. Then, the CA communicates with the AA to analyze the data received from the OO. The CA sends out to AA the pieces of information classified as *country* and *company* as well as the *ID* of the officer that sends the information. The AA classifies the received information as *data*. The CA also sends the *mission* to AA who in turn runs the required analysis, and sends the *analyzed data* to the CA. The PR communicates with the CA to get information regarding a specific *topic*. The CA sends the *mission* related to the topic to PR. Also, CA sends her pieces of information classified as *country*, which are associated with that topic. These pieces of information are classified as *data* at PR. The PR can seek details from the AA. In this case, the AA sends her *data* to PR which are also classified as *data* at PR.

Each communication pattern between two agents can be seen as a protocol by itself. Therefore, for this example, we have four protocols P_1 , P_2 , P_3 , and P_4 . For instance, the protocol P_1 represents the communication pattern between the CA and the OO. The protocol P_2 gives the communication pattern between CA and AA. The protocol P_3 gives the communication pattern between PR and CA and the fourth protocol P_4 gives the communication pattern between PR and AA. In this example, the second protocol should follow the first one and can run in parallel with the third and fourth protocols.

We use the proposed technique to specify the communication between agents and their knowledge. Also, we use it to specify and verify policies regarding the exchange of information among these agents.

4 The Proposed Technique

First, we tackle agent knowledge representation based on information algebra. Then, we use global calculus to represent the communication between agents and link it to their knowledge. Finally, we amend the set of inference rules of Hoare logic by adding new rules and rephrasing the known ones.

The technique assumes that each agent has its own knowledge and information classifications. For example, a piece of information classified as *country* in an agent’s knowledge may be classified as *data* in another agent’s knowledge. In this representation, we remove the restriction of having the same classification in all the knowledges of the system agents. We represent the communication between agents as message passing. Each transmitted message contains: (1) an information that can be composite, and (2) the frames to be assigned to the transmitted information at the receiver’s knowledge.

4.1 Knowledge Representation

In [12], we developed a mathematical structure to specify the agent knowledge and prove that it is an information algebra. The explicit knowledge of an agent is represented by two elements Φ and D . The set Φ consists of pieces of information (we use the words information and piece of information interchangeably) available to the considered agent. There is no restriction on the representation of these pieces of information. They can be represented as formulae as in artificial intelligence literature, functions, etc. In this paper, we represented pieces of information as functions. While D is a lattice of frames such that each piece of information is associated with a frame.

Definition 2 (Agent Information Frame). *Let $\{\mathbb{A}_i \mid i \in I\}$ be a family of sets indexed by the set of indices I and $\mathcal{P}(\mathbb{A}_i)$ be the powerset of \mathbb{A}_i . An information frame D_I is defined as: $D_I \triangleq \prod_{i \in I} \mathcal{P}(\mathbb{A}_i)$ Which can be equivalently written as a set of functions as*

$$D_I \triangleq \{f : I \rightarrow \bigcup_{i \in I} \mathcal{P}(\mathbb{A}_i) \mid \forall(i \mid i \in I : f(i) \in \mathcal{P}(\mathbb{A}_i))\} \quad \square$$

Let $J \subseteq I$ and $\mathcal{I}_J \subseteq I \times I$ such that $\mathcal{I}_J = \{(x, x) \mid x \in J\}$ (i.e., \mathcal{I}_J is the identity on J). Given the frame D_I , we can define D_J as $\{g \mid \exists(f \mid f \in D_I : g = \mathcal{I}_J;f)\}$ where $:$ denotes relational composition. In [11], we prove that $(\{D_J\}_{J \subseteq I}, \gamma, \wedge)$ is a lattice where γ is the join of two frames while \wedge is their meet. For simplicity, we use D to denote the lattice $(\{D_J\}_{J \subseteq I}, \gamma, \wedge)$. On the lattice D and for D_J and D_K frames in D , it is known [5, page 39] that we have $D_J \preceq D_K \Leftrightarrow (D_J \gamma D_K = D_K) \Leftrightarrow (D_J \wedge D_K = D_J)$.

It should be noted that the lattice that we have is different from that of Bell-LaPadula model. Our aim from this lattice representation is to represent frames

of atomic information as in $D_{\{country\}}$ and $D_{\{company\}}$ and to represent frames of composite information as in $D_{\{country, company\}}$.

For the given example, the set of relevant frames of CA is indexed by $I_C = \{officerID, mission, country, company, employee, analyzed-data, topic\}$, the set of frames of OO is indexed by $I_O = \{mission, data\}$, the set of frames of AA is indexed by $I_A = \{mission, data, analyzed-data\}$, and the set of frames of PR is indexed by $I_P = \{topic, mission, data\}$.

An information φ is a function which can be written as a set of 2-tuples (i, A) where i is an index and A is a set. The initial knowledge of the coordinator (Φ^C, D^C) can contain one piece of a composite information φ such that $\Phi^C = \{\varphi\}$ where $\varphi = \{(officierID, \{JohnDo\}), (mission, \{Cobra\}), (topic, \{Economy\})\}$. We denote the domain of an information by using the labelling operator d . The domain of φ is $d(\varphi) = D_{\{officierID, mission, topic\}}$.

Each frame D_J contains a special element called the *empty information* e_{D_J} and is defined as $\{(i, \emptyset) \mid i \in J\}$. Whenever, it is clear from the context, we write e_J instead of e_{D_J} . An information φ is called *atomic* if $\varphi = e_\emptyset$ or $d(\varphi) = D_{\{j\}}$ for $j \in I$. An information can be seen as a row in a table where the table header represents the indices of the frame of the information under that header. An empty information can be seen as a table with only header and e_\emptyset can be seen as empty page that dose not contain even the header. An atomic information can be seen as a one cell of the table or as an empty page.

For the following definitions, let $d(\varphi) = D_J$ and $d(\psi) = D_K$. We define a binary operator \cdot to combine information (we write $\varphi\psi$ to denote $\varphi \cdot \psi$). We use this operator to represent composite information made of pieces of information.

$$\varphi\psi \triangleq \{(i, A) \mid i \in J \cap K \wedge A = \varphi(i) \cup \psi(i)\} \cup \{(i, A) \mid i \in J - K \wedge A = \varphi(i)\} \cup \{(i, A) \mid i = K - J \wedge A = \psi(i)\}.$$

We also define a binary operator $\downarrow: \Phi \times D \rightarrow \Phi$ to extract a part of an information that belongs to a specific frame as $\varphi^{\downarrow D_J} \triangleq \mathcal{I}_{D_J}:\varphi$ where D_J is a frame and φ is an information such that $D_J \in D$ and $\varphi \in \Phi$. The \downarrow operator can be used to extract a specific kind of information. For example, let $\varphi = \{(OfficierID, \{John Do\}), (mission, \{Cobra\}), (topic, \{Economy\})\}$, then $\varphi^{\downarrow D_{\{mission\}}} = \{(mission, \{Cobra\})\}$.

We define a partial order relation \leq on information as $\varphi \leq \psi \Leftrightarrow J \subseteq K \wedge \forall(i \mid i \in J : \varphi(i) \subseteq \psi(i))$ and we say that ψ is *more informative* than φ . This relation indicates whether or not an information is a part of another one. We use this relation to verify the existence of an information in the knowledge of an agent. An information can be in the knowledge of an agent as a part of a composite information. The special element e_\emptyset of D_\emptyset is the least informative information i.e., $\forall(\varphi \mid \varphi \in \Phi : e_\emptyset \leq \varphi)$.

In addition to the information algebra operators, we define an operator to remove a piece of information from another one as follows:

$$\varphi - \psi \triangleq \{(i, A) \mid i \in J \cap K \wedge A = \varphi(i) - \psi(i)\} \cup \{(i, A) \mid i \in J - K \wedge A = \varphi(i)\}.$$

We prove the following proposition by using the definition of the operators and sets identities. We give the proof in [13].

Proposition 1. *Let φ, ψ and χ be pieces of information and let e_J be the empty information on D_J , we have:*

1. $d(\varphi - \psi) = d(\varphi)$
2. $\varphi - e_J = \varphi$
3. $e_J - \varphi = e_J$
4. $\varphi \leq (\psi - \chi) \Rightarrow \varphi \leq \psi$
5. $\varphi \leq \psi \Rightarrow \varphi - \psi = e_{d(\varphi)}$
6. $(\varphi\psi - \psi)^{\downarrow d(\varphi)} \leq \varphi$
7. $\varphi \leq \psi \Rightarrow (\chi - \varphi)\psi = \chi\psi$

□

The proposition gives some properties of the remove operator. Proposition [1\(1\)](#) indicates that removing pieces from an information does not change the frame of that information. Proposition [1\(2\)](#) and [1\(3\)](#) state that removing an empty piece from an information does not affect that information, and removing a piece of information from the empty information gives the same empty information. Also, the proposition relates the more informative relation with the remove operator as shown in Proposition [1\(4\)](#) and [1\(5\)](#). Proposition [1\(6\)](#) and [1\(7\)](#) relate the remove operator with the combine operator.

As we assume that the frame of pieces of information can be changed during their transmission from one agent to another, we define a frame substitution function that substitute a part of a frame of an information with another. We define *frame substitution* as $fs(\varphi, D_J, D_K) \triangleq \varphi^{\downarrow D(L-J)} \cdot (\varphi^{\downarrow D_J} [D_K/D_J])$ where the sets J and K are singleton subsets of the set of indices I and $d(\varphi) = D_L$. For example, let $\varphi = \{(country, \{France\}), (topic, \{economy\})\}$. Then, $fs(\varphi, D_{\{country\}}, D_{\{data\}}) = \{(data, \{France\}), (topic, \{economy\})\}$.

Proposition 2. *Let J and K be singleton subsets of the set of indices I .*

1. $D_J \preceq d(\varphi) \vee \varphi = fs(\varphi, D_J, D_K)$
2. $D_K \preceq d(\varphi) \vee \varphi = fs(fs(\varphi, D_J, D_K), D_K, D_J)$

□

We give the proof in [13](#). The *knowledge* of each agent is modeled as an information algebra $\mathcal{N} \triangleq (\Phi, D)$. Based on the operators of information algebra, we introduce several functions that we use later for specifying communication between agents.

- $isInKnowledge(\mathcal{N}, x, \varphi) \triangleq \exists(\psi \mid \psi \in \Phi : x \in D \wedge \varphi \leq \psi \wedge x \preceq d(\psi))$.
This function verifies the existence of an information in the knowledge \mathcal{N} associated with the frame x and is more informative than φ .
- $extract(\mathcal{N}, x, \varphi) \triangleq \{\psi^{\downarrow x} \mid x \in D \wedge \psi \in \Phi \wedge \varphi \leq \psi \wedge x \preceq d(\psi)\}$.
This function extracts pieces of information from the knowledge \mathcal{N} that contains φ and restricts them to the frame x .

As Φ in \mathcal{N} is a set, the operators on sets can be applied on Φ . For protocol specification, we use the insert and update functions. The insert function $insert(\mathcal{N}, \varphi)$ inserts the information φ into Φ . While the update function $update(\mathcal{N}, \psi, \varphi)$ updates the information ψ with φ in Φ . Formally, $update(\mathcal{N}, \psi, \varphi) \triangleq (\{(\chi - \psi) \cdot \varphi \mid \chi \in \Phi \wedge \psi \leq \chi\} \cup \{\chi \mid \chi \in \Phi \wedge \neg(\psi \leq \chi)\}, D)$. In the insert and update functions, there is always a

condition that $d(\varphi) \in D$. We define the function $choose(\Phi)$ to select a piece of information randomly from Φ . If Φ is empty, it returns the empty information e_\emptyset . We prove the following propositions which help in verifying policies.

Proposition 3. *Let φ and ψ be pieces of information and let \mathcal{N} be a knowledge.*

1. $\varphi \leq \psi \wedge \varphi \leq \chi \Rightarrow update(update(\mathcal{N}, \varphi, \psi), \varphi, \chi) = update(\mathcal{N}, \varphi, \psi \cdot \chi)$
2. $isInKnowledge(\mathcal{N}, d(\varphi), \varphi) \vee update(\mathcal{N}, \varphi, \psi) = \mathcal{N}$

Proof. The detailed proof can be found in [13].

1. The proof applies the definitions of $update$. Also, it uses the set union axiom, the distributivity axiom, the trading rule for \exists , the nesting axiom, Proposition [14], the substitution axiom, and properties of propositional logic.
2. We use the definitions of $update$ and $isInKnowledge$ functions. Also, the proof uses properties of set theory. \square

4.2 Specification of Communications Among Agents

To link global calculus with agent knowledge representation, we explicitly articulate communication, assignment, and conditional terms that involve the knowledge of agents in the context of information algebra.

The *Communication Term* in global calculus is $A \rightarrow B : s\langle op, e, y \rangle$. It is used to express the communication between agents A and B using the channel s , where A is the sender, B is the receiver, op is an operator name used in the communication, and e is an expression evaluated at A whose value is stored in the variable y at B . The operator op does not have a semantics. It is mainly used to have a structured communication between agents and for type checking.

To specify the exchange of information among agents of a system, we represent the expression e as an information φ . We also give a semantics to the operator op which contains a condition c on the receiver knowledge to extract information from the transmitted message and indicate the frame of the information at the receiver knowledge (i.e., frame substitution for the transmitted information). We represent the operator as $op(c, D_J/D_K)$. Therefore, we represent the communication step as $A \rightarrow B : s\langle op(c, D_J/D_K), \varphi, y \rangle$. This step indicates that agent A sends an information φ to agent B . If the condition is satisfied in the knowledge of B , then B applies frame substitution to φ and stores the result in its local variable y . Otherwise, B stores e_\emptyset in y . We can generalize the operator to apply more than one frame substitution at one step. We use in this paper the simple form for clarity.

The Assignment Term: In global calculus, the term $x@A := e$ is used to specify the assignment of the value of the expression e to the variable x located at A . Since we represent the knowledge of agents as an information algebra, the assignment term in our specification becomes $insert(\mathcal{N}^A, \varphi)$ or $update(\mathcal{N}^A, \varphi, \psi)$ depending on the context. The knowledge $\mathcal{N}^A \triangleq (\Phi^A, D^A)$ represents the knowledge of agent A .

The Conditional Term: In global calculus, the term *if* $e@A$ *then* I_1 *else* I_2 is used to specify selecting one of the branches I_1 and I_2 based on the evaluation of the Boolean expression e at the agent A . Since we represent the knowledge of the agents as an information algebra, the expression e is based on the knowledge of agent A and is represented as *isInKnowledge*(\mathcal{N}, x, φ).

The communication between agents presented in the example can be specified in global calculus as $(P_1; P_2)|P_3|P_4$. The parallel operator between two terms is equivalent to all possible interleaving between their steps. Due to space limitation, we give the specification of the third protocol only.

```

1  PR → CA : ch(vs).
2  PR → CA : s( op({true}, D_{topic}/D_{topic}),
                choose(extract(D_{topic}, {(topic, {economy})}), N^{PR})
                y).
3  CA → PR : s( op({true}, D_{data}/D_{country}),
                choose(extract(D_{mission, country}, y, N^{CA})
                x).
4  update PR {(topic, {Economy})} {(topic, {Economy})} · x.
5  0
    
```

The first step represents initiating a session between PR and CA. The second step represents sending a message through the channel s . This message contains a topic. The third step specifies sending a composite information associated with the mission and country frames. The message indicates through the operator that the frame *country* is to be substituted with the frame *data* at PR. The condition for extracting the information is *true* i.e., no condition. The last step specifies updating the knowledge of PR.

4.3 Analysis

We analyze security policies in multi-agent systems by using an amended version of Hoare logic. In the proposed technique, a *policy* is a predicate on the knowledges of a set of agents. A precondition P in the Hoare triple $\{P\}S\{Q\}$ represents the initial knowledge of agents, S represents the specification of a communication protocol expressed in global calculus and information algebra, and the postcondition Q represents the negation of a policy on the knowledge of the considered agents. The postcondition is expressed as a predicate articulated using terms of the language of information algebra.

To verify a policy, we first calculate the weakest precondition of the protocol based on the postcondition (negation of a policy) and protocol specification. Then, we evaluate the term $P \Rightarrow wp(S, Q)$. If the evaluation is true, then the policy is not satisfied, otherwise, it is satisfied.

The inference rules of Hoare logic are generally articulated in terms of primitives of a programming language. In Figure 2, we rephrase some of these rules in terms of the language of information algebra and global calculus.

The (INACTION) and (INIT) terms are considered to be equivalent to the skip program as they do not have any effect on the knowledge of the communicating agents. The (INSERT) and (UPDATE) terms are considered to be equivalent to the assignment statement. Therefore, their weakest precondition is

$$\begin{array}{ll}
\text{(INACT)} \frac{-}{\{P\} 0 \{P\}} & \text{(COMM)} \frac{\{P \wedge C\} \ y@B := fs(\varphi, D_J, D_K)\{Q\} \ \{\neg P \wedge C\} \ y@B := e_0\{Q\}}{\{P\} A \rightarrow B : s(op(C, D_J/D_K), \varphi, y)\{Q\}} \\
\text{(INIT)} \frac{-}{\{P\} A \rightarrow B : ch(v\bar{s}) \{P\}} & \text{(UPDATE)} \frac{-}{\{P[\mathcal{N}^A / \text{update}(\mathcal{N}^A, \varphi, \psi)]\} \ \text{update}(\mathcal{N}^A, \varphi, \psi) \{P\}} \\
\text{(INSERT)} \frac{-}{\{P[\mathcal{N}^A / \text{insert}(\mathcal{N}^A, \varphi)]\} \ \text{insert}(\mathcal{N}^A, \varphi) \{P\}} & \text{(COMP)} \frac{\{P\} S \{Q\}, \{Q\} T \{R\}}{\{P\} S; T \{R\}}
\end{array}$$

Fig. 2. Weakest precondition inference rules for the verification of information flow

substituting the agent knowledge \mathcal{N}^A with $\text{insert}(\mathcal{N}^A, \varphi)$ or $\text{update}(\mathcal{N}^A, \varphi, \psi)$. The protocol step composition is the same as program composition. Therefore, the (COMP) inference rule is equivalent to the weakest precondition of program composition. The (COMM) contains a condition to extract the transmitted information and a frame substitution function. If the condition is satisfied, then the frame substitution function is applied to the information and the result is stored in the variable y located at the receiver. Otherwise, the empty information is stored in y .

In the analysis, we are taking the initial knowledge of agents into consideration which plays an important role as shown by the following propositions.

Proposition 4. *let $\mathcal{N} = (\{e_0\}, D)$ be an empty knowledge. We have:*

1. $\text{update}(\mathcal{N}, \varphi, e_0) = \mathcal{N}$
2. $\text{insert}(\mathcal{N}, e_0) = \mathcal{N}$
3. $\text{choose}(\text{extract}(\mathcal{N}, x, \varphi)) = e_0$

Proof. The detailed proof can be found in [13].

1. The proof applies the definition of update, Proposition 1(3), singleton membership, set union axiom and properties of propositional logic.
2. The proof applies the definition of insert and the idempotency of \cup .
3. The extract function would either return an empty set or a set with one element e_0 . In both cases, *choose* will return e_0 . \square

As a consequence result of Proposition 4, we can prove the following proposition:

Proposition 5. *Let $\mathcal{N}^A = (\{e_0\}, D^A)$ and $\mathcal{N}^B = (\{e_0\}, D^B)$ be the knowledges of two agents communicating through a protocol S . We have $wp(S, Q) \Leftrightarrow Q$.*

Proof. The postcondition Q is on the knowledge of agents. According to Proposition 4, the knowledge of agents with empty knowledge does not change through the insert and update functions. \square

Proposition 5 states that if the initial knowledge of agents A and B contain only the empty information, then from an information flow prospective, protocol specification is equivalent to skip. On the other side, if the initial knowledge of agents A and B contains all possible information i.e., $\Phi = \{\varphi \mid \varphi \in D_J \text{ for } J \subseteq I\}$, then we can prove the following proposition:

Proposition 6. *let $\mathcal{N} = (\Phi, D)$ where $\Phi = \{\varphi \mid \varphi \in D_J \text{ for } J \subseteq I\}$. We have $\text{insert}(\mathcal{N}, \varphi) = \mathcal{N}$*

Proof. The proofs uses the definition of insert and the fact that $\varphi \in \Phi$. \square

Proposition 7. *Let $\mathcal{N}^A = (\Phi, D^A)$ and $\mathcal{N}^B = (\Phi, D^B)$ where $\Phi = \{\varphi \mid \varphi \in D_J \text{ for } J \subseteq I\}$ be the knowledge of two agents communicating through a protocol S . Also, assume that S does not contain an update. We have $\text{wp}(S, Q) \Leftrightarrow Q$.*

Proof. The postcondition Q is on the knowledge of agents. According to Proposition 6, the knowledge of agents does not change through the insert function. \square

We exclude the update function from Proposition 7 because it does not preserve the knowledge. The update function removes and combines a piece of information to the information that already exists in the knowledge of an agent.

For the given example, we can verify several policies such as (1) the AA should not know pieces of information classified as *employee* in the CA knowledge (2) the PR should not know a piece of information classified as *company* in the CA knowledge (3) the PR should not know the *mission* of an *officer ID*. Due to space limitation, we focus on the second policy and refer the reader to [13].

To analyze a policy, we specify the communication S between agents, the precondition P , and the postcondition Q . Then, we prove that $p \Rightarrow \text{wp}(S, Q)$. We specify the communication between agents by using global calculus and information algebra as shown in Section 4.2. The precondition is a predicate on the initial knowledge of agents represented as a conjunction of formulae of the form $\text{isInKnowledge}(\mathcal{N}, x, \varphi)$. For example, to indicate that the initial knowledge of the coordinator contain the information $\varphi = \{(\text{OfficierID}, \{\text{JohnDo}\}), (\text{mission}, \{\text{Cobra}\}), (\text{topic}, \{\text{Economy}\})\}$, we use the predicate $\text{isInKnowledge}(\mathcal{N}^C, d(\varphi), \varphi)$. The postcondition, which is a predicate on the knowledge of agents, is the negation of a policy. The postcondition of the second policy stated above is

$$\begin{aligned} \exists(\varphi, x \mid x \in D^{PR} \wedge \varphi \in D_{\{\text{company}\}}^{CA} : \varphi \neq e_{\{\text{company}\}} \wedge \\ \text{isInKnowledge}(\mathcal{N}^{CA}, D_{\{\text{company}\}}, \varphi) \wedge \\ \text{isInKnowledge}(\mathcal{N}^{PR}, x, \text{fs}(\varphi, D_{\{\text{company}\}}, x))) \end{aligned}$$

We developed a prototype tool which finds the weakest precondition from S and Q . Then, the tool writes it using the syntax of PVS language. After that, we use PVS to prove that $P \Rightarrow \text{wp}(S, Q)$. Currently, we are proving theorem in PVS through the interactive mode. However, we intend to automate this step. The proof is based on the propositions defined in this paper. We are able to prove that $P \Rightarrow \text{wp}(S, Q)$ for the second policy which means that it contains a flaw. Indeed, the policy is not satisfied because PR gets an information from AA which is classified as *company* in the CA knowledge. More details on the proof of this policy can be found in the technical report [13]. If we are unable to prove $P \Rightarrow \text{wp}(S, Q)$ for a policy, this does not mean that the policy is satisfied. To prove that a policy is satisfied, we should prove $\neg(P \Rightarrow \text{wp}(S, Q))$.

5 Discussion and Related Work

In this section, we compare the proposed framework with Bell-LaPadula and Chinese Wall models. Then we present the aspects that distinguish the proposed framework from the existing techniques.

5.1 Bell-LaPadula Model

We compare Bell-LaPadula model [2] with the proposed technique regarding the following aspects: (1) *Security Level*: Bell-LaPadula model assigns security levels to subjects and objects. The security level forms a lattice which enables us to have a relation as: *private* is more sensitive than *public*. In the proposed technique, we use a lattice of frames to classify information. Our lattice enables us to perform analysis on a composite information. We can give security levels to information or agents through policies. (2) *Objects*: In Bell-LaPadula model, objects cannot be decomposed or combined. While in the proposed technique they can be combined (i.e., combining information). (3) *Rules*: Bell-LaPadula model defines rules for reading and writing into objects. These rules can be specified within the proposed technique as the condition in the communication step.

5.2 The Chinese Wall Model

The Chinese Wall [3] is a security model with three levels of significance. The lowest level contains pieces of information. Pieces of information of the same cooperation are grouped into a company dataset. Company datasets whose cooperation are in competition are grouped into a conflict of interest class. An agent should have access only to information of one company dataset of the conflict of interest class.

The Chinese Wall can be represented within the proposed technique as follows. The pieces of information of Chinese Wall model correspond to the pieces of information within the proposed technique, while the company datasets correspond to frames. We do not have a direct representation of the conflict of interest class. However, this is a policy which can be stated within the proposed technique in different ways. One way is to state it as the condition in the communication step i.e., an agent can receive an information only if the knowledge of that agent does not contain information belonging to frames which are in conflict with the received information frame.

5.3 Verification Techniques

Bell-LaPadula and Chinese Wall are two models that specify policies for a secure information flow between agents. In the literature, there exists several techniques [1,6,8,14] to verify that the information flow between agents satisfy predefined policies. The proposed technique has the following aspects which distinguish our work from others:

Analyzing a composite information flow: An advantage of the proposed technique over the existing ones is analyzing composite information flow. The technique can analyze the ability of an agent to link pieces of information together from different resources. Existing techniques such as SPA [6] verify only if an agent is able to get an information that it is not supposed to have.

Specifying agent knowledge: Agent knowledge representation adopted in this paper enables specifying the evolution of knowledge through communication. This specification is different from using variables that their new value substitute the old one (i.e., there is no evolution). Also, each agent has its own lattice of frames that classifies the information it holds. We remove the restriction that all pieces of information should have the same classification in agents' knowledge. Removing this restriction provides flexibility in specifying heterogeneous subsystems.

Analyzing distributed systems: The technique can specify distributed systems. By using global calculus, we are able to compose communication steps either concurrently or sequentially.

Analyzing system at design level: The technique enables the analysis of systems at design level. Analyzing systems at an early stage of the software development life cycle and an early detection of a system security vulnerability would reduce the cost of addressing it. This stage of analysis is different from typing systems [8,9,15] that have been widely used for analyzing information flow at the implementation level.

6 Conclusion

In this paper, we propose a technique for the analysis of information flow between agents in multi-agent systems. We develop an algebraic structure to specify the knowledge of agents based on information algebra. We also link global calculus with information algebra and use them to specify communication protocols. Furthermore, we use an amended version of Hoare logic to analyze the information flow between agents. We report on a prototype tool to derive the weakest precondition of a communication protocol. Then, we verify whether the initial knowledge of agents implies the weakest precondition by using the PVS theorem prover.

The proposed technique provides a comprehensive language to specify agents knowledge and their communications. For example, information algebra allows reasoning on a composite information as in the third policy of the given example. The global calculus allows specifying a composition of protocols using the sequential and parallel operators. Also, the analysis takes the initial knowledge of agents into consideration which affects the satisfiability of a policy. Finally, performing the analysis before implementing the system allows for detecting flows at an early stage. The implementation can be generated automatically from the specification which can be a future work. However, the trade off for having a detailed analysis is increasing its complexity especially in large systems.

References

1. Alghathbar, K., Farkas, C., Wijesekera, D.: Securing UML information flow using FlowUML. *Journal of Research and Practice in Information Technology* 38(1), 111–120 (2006)
2. Bell, D.E., La Padula, L.J.: Secure computer system: Unified exposition and multics interpretation. Technical Report ESD-TR-75-306, The MITRE Corporation (March 1976)
3. Brewer, D.F.C., Nash, M.J.: The Chinese Wall security policy. In: *IEEE Symposium on Security and Privacy*, May 1989, pp. 206–214 (1989)
4. Carbone, M., Honda, K., Yoshida, N.: Structured communication-centred programming for web services. In: De Nicola, R. (ed.) *ESOP 2007*. LNCS, vol. 4421, pp. 2–17. Springer, Heidelberg (2007)
5. Davey, B.A., Priestley, H.A.: *Introduction to Lattices and Order*. second edition. Cambridge University Press, Cambridge (2002)
6. Focardi, R., Gorrieri, R.: The compositional security checker: A tool for the verification of information flow security properties. *IEEE Transactions on Software Engineering* 23(9), 550–571 (1997)
7. Hoare, C.A.R.: An axiomatic basis for computer programming. *Communications of the ACM* 12(10), 576–580 (1969)
8. Hristova, K., Rothamel, T., Liu, Y.A., Stoller, S.D.: Efficient type inference for secure information flow. In: *PLAS 2006: Proceedings of the 2006 workshop on Programming languages and analysis for security*, pp. 85–94. ACM, New York (2006)
9. Kobayashi, N.: Type-based information flow analysis for the π -calculus. *Acta Informatica* 42(4), 291–347 (2005)
10. Kohlas, J., Stärk, R.F.: Information algebras and consequence operators. *Logica Universalis* 1(1), 139–165 (2007)
11. Sabri, K.E., Khedri, R.: A mathematical framework to capture agent explicit knowledge in cryptographic protocols. Technical Report CAS-07-04-RK, department of Computing and Software, Faculty of Engineering, McMaster University (October 2007), <http://www.cas.mcmaster.ca/cas/0template1.php?601> (accessed January 19, 2008)
12. Sabri, K.E., Khedri, R., Jaskolka, J.: Specification of agent explicit knowledge in cryptographic protocols. In: *CESSE 2008: International Conference on Computer, Electrical, and Systems Science, and Engineering*, Venice, Canada, October 2008, vol. 35, pp. 447–454. World Academy of Science, Engineering and Technology (2008)
13. Sabri, K.E., Khedri, R., Jaskolka, J.: Automated verification of information flow in agent-based systems. Technical Report CAS-09-01-RK, department of Computing and Software, Faculty of Engineering, McMaster University (January 2009), <http://www.cas.mcmaster.ca/cas/0template1.php?601> (accessed January 19, 2009)
14. Varadharajan, V.: Petri net based modelling of information flow security requirements. In: *Computer Security Foundations Workshop III*, pp. 51–61 (June 1990)
15. Volpano, D., Irvine, C., Smith, G.: A sound type system for secure flow analysis. *Journal of Computer Security* 4(2-3), 167–187 (1996)

A Legal Perspective on Business: Modeling the Impact of Law

Sepideh Ghanavati¹, Alberto Siena², Anna Perini², Daniel Amyot¹, Liam Peyton¹, and Angelo Susi²

¹ SITE, University of Ottawa, Canada

{sghanava, damyot, lpeyton}@site.uottawa.ca

² Fondazione Bruno Kessler - Irst (FBK - Irst)

{siena, perini, susi}@fbk.eu

Abstract. Modern goal-oriented requirements engineering frameworks use modeling as a means of better understanding a domain, leading to an overall improvement in the quality of the requirements. Regulations and laws impose additional context and constraints on software goals and can limit the satisfaction of stakeholder needs. Organizations and software developers need modeling tools that can properly address the potential deep impact legal issues can have on the effectiveness of business strategies. In this paper, we perform a preliminary study into the development of a modeling framework able to support the analysis of legal prescriptions alongside business strategies. We demonstrate, via an example drawn from a case study of the Health Insurance Portability and Accountability Act (HIPAA), how models of this law can be built with the GRL modeling language and how they can be evaluated as part of the business goal models.

Keywords: Business Modeling, Goal-oriented Requirement Language, HIPAA, Law Modeling.

1 Introduction

In the development of modern information systems, understanding and analyzing the purpose of a software system before defining its desired functionality is becoming more and more important [6]. The early requirements analysis phase [9] is the part of the software development process in which it is possible to effectively understand the justification and need for a new system with respect to the organizational setting in which it will operate. In this phase, the needs of the stakeholders are taken into consideration. The business strategies define possibly the most important needs of an organization, and are likely the driving force behind the development initiative [15]. In this context, the goal-oriented techniques proposed in the last decade [14][10] try to answer software *why* questions in addition to the standard *what* and *how* as they relate to system functionality. Answers to *why* questions will ultimately link software requirements to stakeholder needs, preferences and objectives.

Goal analysis techniques [3] are useful in order to understand the structure and the correlations between goals, their decomposition into more fine-grained sub-goals, and

their relationship with operational plans. Moreover, reasoning techniques applied to goal models [5] can be very useful in verifying model properties and have shown to be useful for analysts in conflict resolution processes that weigh the different aspects of a strategy [15] and work to improve the decisions made about the models.

Although these techniques are very helpful when reasoning about models with a single goal, they are inadequate in the support of strategic decisions of a real multi-goal organization. This is mainly due to the fact that they are typically built from the perspective of the system stakeholders and do not consider other dimensions like the impact of laws on the organizational strategy. For example, ignoring legal prescriptions may lead to a set of requirements that is fully aligned to stakeholders needs, but that is in violation of the law. Such a situation will eventually result in the need for later correction of the requirements and increased costs.

In recent years, there have been some attempts to apply goal modeling techniques to the setting of legal documents. However, there are still no complete guidelines for how this can be performed. Furthermore, no analysis has ever been done in order to measure the precision and completeness of any such model.

In this paper, we analyze the capabilities and limitations of the Goal-oriented Requirements Language (GRL) [13] for legal documents. In addition, we discuss the impact of these models on an organization in terms of the possible decisions an organization can make to satisfy the law and still accomplish their business objectives.

The paper is structured as follows: Section 2 discusses work related to legal goal modeling and its impact on business processes. Section 3 details the steps that need to be taken to model legal documents whereas Section 4 illustrates these steps and analyzes the pros and cons of different goal models with the help of a case study affected by the Health Insurance Portability and Accountability Acts (HIPAA). Finally, in Section 5, we analyze the impact of a legal model on an organizational goal model and in Section 6 we present our conclusions.

2 Related Work

In recent years, research has been undertaken to investigate the role of laws in software requirements engineering processes. Antón and Breaux, in [1], developed a systematic process for extracting rights and obligations (including auxiliary concepts such as actors and constraints) from legal texts thereby generating a formal model of law. This work provides an important example of the scale and complexity involved when dealing with the vagueness of legal sources.

In terms of organized legal concepts, we have the LRI-Core [2]. This describes a layered legal ontology, built from a foundational ontology that is instantiated as a domain ontology. LRI-Core is based on the idea that the law is driven by common world concepts and words, and as such the ontology defines concepts such as agent, action and organization together with legal concepts.

The need for legal modeling is also implicitly contained in other works that strive to perform requirements modeling. Darimont and Lemoine use KAOS to model objectives extracted from regulatory texts [4]. Such approaches are based on the similarities between regulations and requirements. Other similar approaches come in the form of the techniques adopted in [7], where a goal model is used to model the goals

and activities prescribed by laws. Ghanavati et al. in [7] built their work on the intuition that there is a need to use the same modeling notation for both the regulations and the organizational processes. As such, traceability links have been established between the law and the business processes, thus enabling the management of their evolution. The special consideration needed to handle evolving requirements and laws are described in [8].

Other frameworks include the Normative i^* framework [16] and Secure Tropos [11]. The Normative i^* framework allows for modeling laws inside an intentional framework and produces effective additions to the requirements system. Secure Tropos constitutes a security-enhanced version of the Tropos methodology and introduces the concepts of service ownership and delegation. In order to ensure access control, strategic dependencies are refined with the conditions of permission and commitment.

With respect to these approaches, this paper studies the modeling of law and its limitations, as well as the impact that laws have on business strategies, and attempts to do so in a quantitative manner.

3 Legal Modeling

Generally speaking, laws deal with prescribing how the world *should* be and as such relate to their deontics. Moreover, laws are very complex artifacts and are hard to capture because they are expressed in natural language and are intentionally vague in order to support as of yet unseen circumstances. There are also the implications of case law which constitute instances of applied law and serve to refine written laws and dictate how they should be interpreted. Requirements modeling languages and processes are not intended to capture this level of complexity. Currently there is no precise approach in dealing with this problem from the literature.

The modeling of legal concepts has gained the attention of some researchers. In [7] the authors suggest to model legal concepts with Goal-oriented Requirements Language (GRL), which is part of the User Requirements Notation (URN). However, they ignore the possibility of using Use Case Maps (UCM), another complementary notation of URN for modeling laws and regulations since they believe that usually legal documents do dictate business processes. URN is a new ITU-T recommendation used to help capture, model, and analyze user requirements in the early stages of design. GRL is a goal modeling notation based on i^* and the Non-Functional Requirements (NFR) framework. It includes intentional elements (i.e. *goals*, *softgoals*, *tasks* and *resources*) as well as different links which connect these elements to each other. These links represent different types of relationships such as contribution, correlation, decomposition and dependency. UCM focuses mainly on the functional requirements of a system and the causal relationships between the responsibilities of different use cases. UCM can be used to model business processes as well as to capture, elicit and validate use cases. With these two complementary views, URN also allows for the alignment of business goals and business processes.

The benefit of using a common goal-oriented language such as GRL is that the *desired* behavior of actors (the object of legal prescriptions) can be modeled using the

same language as the actual behavior. It is helpful that GRL is graphical so that it can be easily analyzed and discussed.

In fact, legal documents address the concepts of *actors*, *rights*, *obligations* and *constraints* [1]. According to Breaux et al., rights are claims that are assigned to the right bearer whereas obligations are the responsibilities applied to the obligated party and that he must fulfill in order to comply with the regulation. When an actor has a right, he has the (legal) capability to use the right to accomplish his goals. Conversely, when an actor is charged with an obligation he has the (legal) responsibility to perform the corresponding prescribed actions. It follows that the holders of rights and obligations are complementary because in order for an actor to apply a right, another actor has to perform an activity. Finally, constraints can articulate both rights and obligations which can be represented as precondition activities, exceptions or dependent obligations and rights. Therefore, a legal prescription can be described as an obligation or right statement with the provision of optional constraints.

This definition of rule statements can help us model them with GRL. Obligations, as mentioned above, are activities which must be performed by an actor in order to satisfy the goals that are described as rights attributed to another actor. In this case, there are some goals to be achieved, some tasks required to satisfy them and some actors bound to the related tasks. If the statement only contains simple rights and obligations, we can model its different parts with GRL intentional elements (soft-goals, goals, tasks) and actors. If the statement contains any constraints, the constraint itself needs to be analyzed separately. Precondition activities can be modeled as tasks in GRL and linked to the related goals or tasks via contribution links of type *make*, *help*, *some positive*, *unknown*, *some negative*, *hurt*, or *break*.

If a constraint contains exceptional situations or actors (which usually represent nested “if” conditions), it cannot be shown with GRL, since they usually have a procedural nature while GRL is only able to depict static representations of goal models. To model these exceptions, we can use UCM instead since this notation is able to model business processes. The benefit of using UCM over other business processing modeling notations is that it has the ability to link its elements to GRL elements. In other words, tasks and actors in GRL can be linked to responsibilities and components in UCM maps.

4 HIPAA Case Study

4.1 Model of the Law

The Health Insurance Portability and Accountability Act (HIPAA) was enacted in the USA in 1996 [12]. HIPAA includes two titles. Title 1 protects health insurance coverage for workers and Title 2 enforces the establishment of national standards for electronic health care transactions and national identifiers for providers, health insurance plans and employers. In addition, Title 2 addresses the privacy and security of health data. As an example to illustrate our work, we will consider Article §164.314 of HIPAA. This article describes “Organizational Requirements” and the contract between the covered entity and the business associate.

Article §164.314. a(1) I prescribes that “A covered entity is not in compliance with the standards in §164.502(e) and paragraph (a) of this section if the covered entity knew of a pattern of an activity or practice of the business associate that constituted a material breach or violation of the business associate’s obligation under the contract or other arrangement, unless the covered entity took reasonable steps to cure the breach or end the violation, as applicable, and, if such steps were unsuccessful (A) Terminated the contract or arrangement, if feasible; or (B) If termination is not feasible, reported the problem to the Secretary.”.

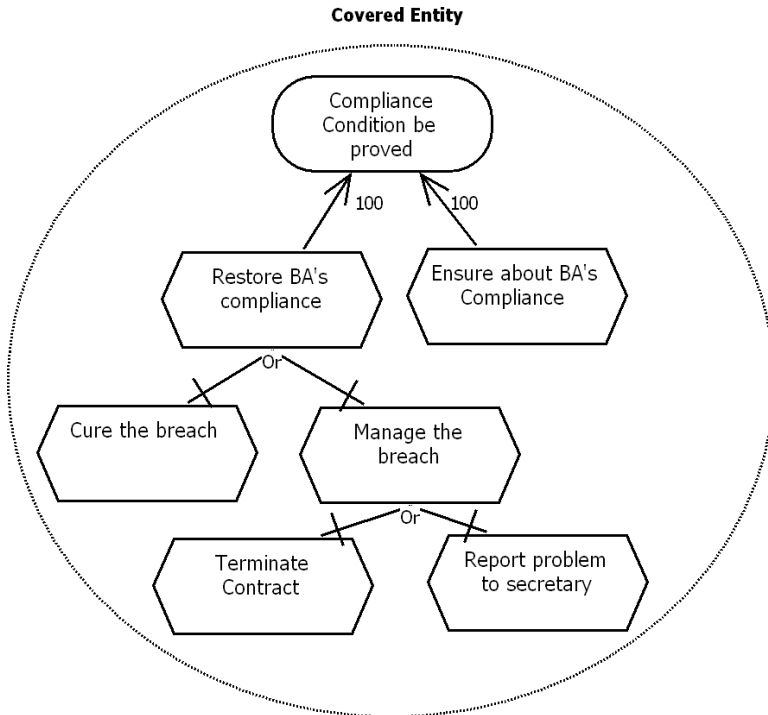


Fig. 1. GRL actor model (CE) of the Article §164.314

The main objective of this part of the article is that the covered entity complies with the standard. It introduces some key examples of the concepts we will use. The paragraph references the two role-playing actors involved, namely the Covered Entity (CE) and the Business Associate (BA). It defines the statement, “knew of a pattern of an activity and/or practice an activity which leads to material breach or violation” as a precondition for “not being in compliance.” This precondition can be modeled as a task which contributes negatively to the main objective. In other words, the BA is compliant with the standard if “the CE does not recognize any pattern of non-compliance.” Another constraint defined in the article is that “if the CE took reasonable steps to cure the breach or end the violation, the BA’s compliance will be restored.” Therefore the task Cure the Breach or End the Violation will help to restore

compliance. However, if it is not possible to cure the breach, the CE has to either Terminate the Contract or Report Problem to Secretary (another condition/constraint). These two tasks also contribute positively to the Restore the Breach activity. Since these two tasks are part of one condition (i.e. *if such steps were unsuccessful*), we put them as sub-tasks of the main task Manage the Breach. This portion of the article can be modeled in GRL. Figure 1 shows an excerpt of the GRL model for Article §164.314.

As mentioned above, GRL intentional elements are *softgoals*, *goals*, *tasks* and *resources*. A softgoal shown as cloud is a kind of goal that can never be concretely achieved. This type of goal represents a high-level goal of the system. In Article §164.314 we have not identified any softgoals. Goals represent the conditions which must be achieved with certainty. Here, the goal of the system is Compliance Condition be Proved. Both softgoals and goals are decomposed until they are operationalized into tasks. Tasks represent the operational solutions to the system and are shown as hexagons. Tasks are usually decomposed with AND/OR links into several sub-intentional elements. Tasks are easily identified as hexagons. For example in Figure 1, Restore BA’s Compliance is a task which has been decomposed by an OR link into two other tasks Cure the Breach and Manage the Breach. Figure 2 shows a brief summary of the GRL notation elements.

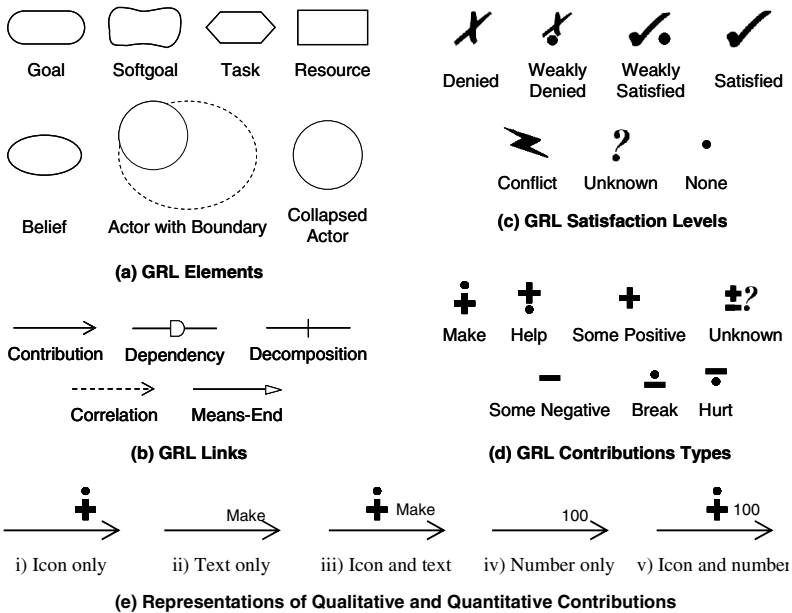


Fig. 2. GRL notation summary

Contribution links can also have different levels of effect on the softgoals or goals connected to them. Links of type *make*, *help*, and *some+* indicate positive relationships that are sufficient, insufficient and unknown respectively. Those links that are labeled as *break*, *hurt* and *some-* are used for negative contributions that are respectively

sufficient, insufficient and unknown. In Figure 1, both tasks Restore the Breach and Ensure about BA’s Compliance have *make* contributions to the goal Compliance Condition be Proved.

The GRL model we built here has some assumptions as well as some limitations. As mentioned above, there are some tasks and activities in the GRL model such as Restore the Breach which are not explicit in the text. However, since the GRL model aims to lessen the complexity of the text, it is necessary to include some intentional elements which are only implied by the text. In addition, there are some situations in the legal text which imposes sequence and priorities for activities. For example in Article §164.314, it is written that “*if such steps were unsuccessful (A) Terminated the contract or arrangement, if feasible; or (B) If termination is not feasible, reported the problem to the Secretary*”. Although, these alternatives are illustrated in the GRL model, their sequence and priorities are not. In this case, we use UCM because this notation is able to model scenarios and business processes. Figure 3 illustrates a UCM which serves to model the “*if conditions*” in the legal document.

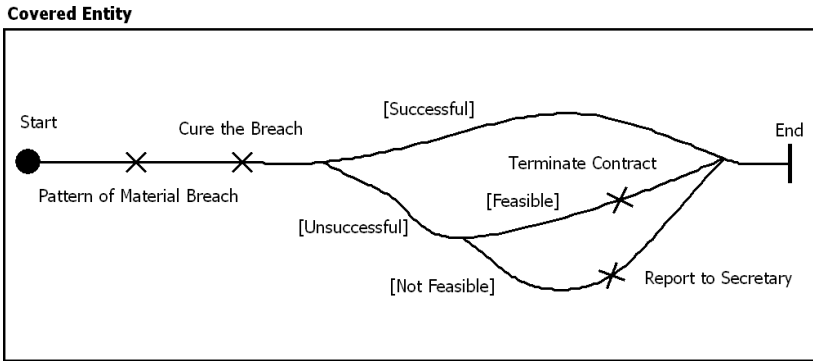


Fig. 3. UCM model of the Article §164.314

Figure 3 includes only one Use Case Map with one component called Covered Entity. This component has a link to the actor CE. There is a path from Start to End with two OR-forks to show the different possibilities. Responsibilities such as Cure the Breach that are mapped to tasks are shown by an ‘X’. Conditions, such as [successful] are shown as bracketed strings.

This UCM shows in case of a breach the Covered Entity has to Cure the Breach and if this activity is not successful and if terminating the contract is feasible, the Covered Entity will Terminate the Contract, otherwise he will Report to the Secretary. By linking the related parts of this UCM map to the above GRL model, we can represent the ordering of the tasks which is critical in any analysis where dependencies exist.

4.2 GRL Model of Law: Pros and Cons

GRL is a conceptual modeling tool that allows goal-based models to be built from legal prescriptions and serves to lessen the complexity of the representation of the

law. To achieve this result, we exploit the fact that legal documents contain goals that are mandatory for the addressed subjects to achieve, and activities that are the legal means for achieving these goals. As a result, we can express in GRL both the goals of the organization and the goals of the law. In addition, since GRL is tied to UCM which is used for business process modeling, the goal model of the law can impose some new requirements on the business process of the organization. However, not everything in the law can be modeled by GRL only. For example, priorities between two options cannot be shown in GRL. In this case, UCM is added to the model to show the process and priorities.

5 Impact Analysis

5.1 Model of the Organization

Ghanavati et al. in [7] mention that in order to have good traceability and be able to manage organizational compliance with law, it is beneficial to build a corporate goal model using the same notation as the model of law. As a result, the organizational model is also built with GRL. In this section, we aim to analyze how the legal goal model impacts the satisfaction of the organization's main objective. Legal prescriptions can impact the same goal differently given two scenario alternatives. In order to analyze the relative degree of impact, we can use GRL evaluation strategies. These have the capability to analyze goals quantitatively, qualitatively or as a mixture of both approaches.

The quantitative evaluation algorithm uses quantitative contributions, quantitative degrees of goal satisfaction and quantitative importance values for actors, all in the range of -100 to +100. An evaluation algorithm then propagates the effect of each into a single numerical result also in the same range of -100 and +100. The qualitative evaluation algorithm uses qualitative contribution values (i.e. *make, some positive, help, none, hurt, some negative, break*), qualitative degrees of satisfaction (i.e. *denied, weakly denied, weakly satisfied, satisfied, conflict, unknown, none*) and qualitative importance values (i.e. *high, medium, low, none*). Finally, the mixed evaluation algorithm includes both quantitative and qualitative values at the same time. All of these algorithms follow a bottom-up approach. In this example, drawn from our HIPAA case study, we use the quantitative evaluation strategy.

Since the legal document we are using in our case study is HIPAA, the organization that it affects is a healthcare organization. In our example, we analyze the task of disclosing Protected Health Information (PHI) to different users in different scenarios and its impact on achieving the main goal of the hospital which is to Improve the Quality of Healthcare. In our case, the hospital aims to Disclose the PHI to Researchers or Disclose the PHI to the Healthcare Assistants. Information is disclosed to the researcher in order to help him satisfy his objective which is to Do Research. Figure 4 shows an excerpt of the associated GRL goal model.

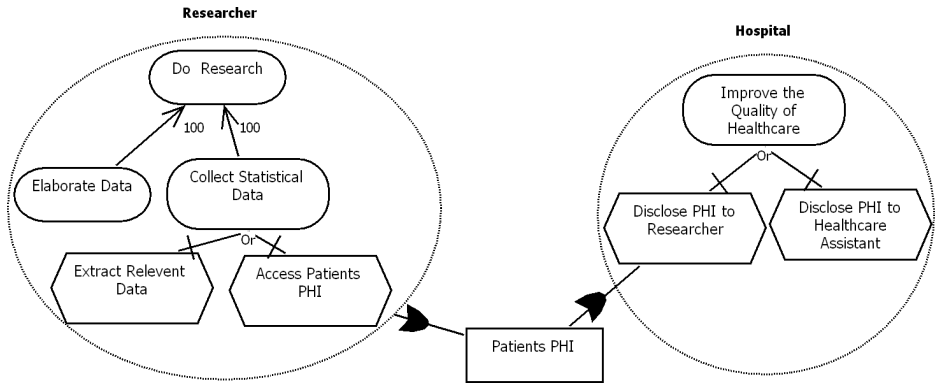


Fig. 4. Organization GRL Model

5.2 The Organization Model Against the Law Model: A Business Analysis

As an example of how to analyze how the GRL model of the law affects the objectives of researchers and healthcare assistants, we define two basic scenarios. We selected these two scenarios to illustrate how modeling of law by GRL can help the organization to analyze the satisfaction of their goals. These scenarios are based on Article §164.506 which states that, *except with respect to uses or disclosures that require an authorization under §164.508(a)(2) and (3), a covered entity may use or disclose protected health information for treatment, payment, or health care operations as set forth in paragraph (c) of this section, provided that such use or disclosure is consistent with other applicable requirements of this subpart.* Article §164.508 specifies the exception that Article §164.506 applies, *except as otherwise permitted or required by this subchapter, a covered entity may not use or disclose protected health information without an authorization that is valid under this section.*

Figure 5 illustrates the first scenario where a healthcare assistant wants access to PHI. In this Figure, the task Disclose PHI to Healthcare Assistant gets the value of 100. By selecting this task, the goal of the hospital, which is Improve the Quality of Healthcare, is satisfied (value of 100). According to the Article §164.506, the Covered Entity can give the permission to disclose PHI for healthcare operations, Healthcare Operation Request, which is selected with a value of 100. Therefore, Disclosing PHI to Healthcare Assistant can satisfy the main objectives of both hospital (i.e. Improve the Quality of Healthcare) and the covered entity (i.e. Disclose PHI).

In the second scenario, the Researcher wants to get access to PHI. Therefore, the task Disclose PHI to Researcher is selected and the goal Improve the Quality of Healthcare is satisfied (both shown with the value of 100). According to Article §164.508, without the authorization the researcher cannot get access to PHI. Therefore, if the covered entity discloses data to researcher (i.e. the task Other Users' Request is satisfied), the objectives of the researcher and the hospital are satisfied but the hospital will be in breach of the law. Figure 6 illustrates this situation. The goal Disclose PHI is *denied* (unsatisfied) as it has a satisfaction level of -100. In order to comply with HIPAA, the hospital should prohibit disclosure of PHI to the researcher. However, this situation results in an unsatisfied goal of researcher and the hospital.

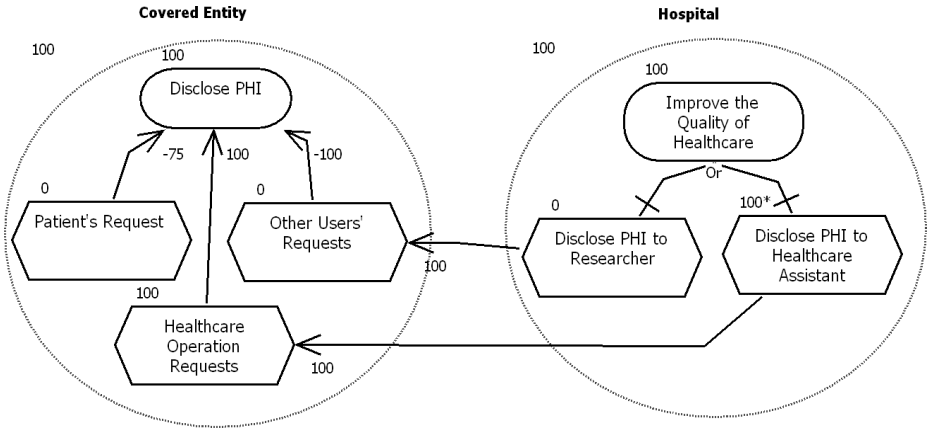


Fig. 5. Disclosing PHI to Healthcare Assistant

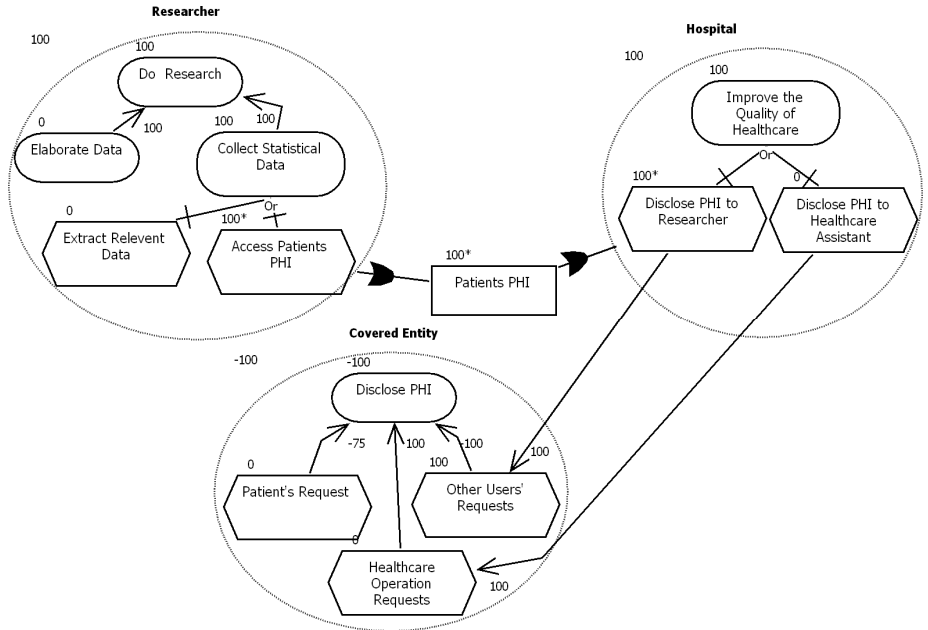


Fig. 6. Disclosing PHI to the Researcher

This impact analysis illustrates how a goal model of the law can help organizations make decisions, perform trade-off analysis, and better understand how they can achieve their internal goals and still comply with the law under which they must operate.

6 Conclusions

In this paper we have shown how the goals of an organization are affected by the law using different scenarios drawn from a case study of HIPAA. These scenarios highlighted how a modeling approach, like the one proposed here, can help formalize an organization's approach to dealing with conflicting objectives that may involve potential violations of legal responsibility. To deal with these conflicting objectives, we discussed the need for a formal goal model of the law and the necessary steps in order to create it. Such a modeling is possible if one leverages the combined capabilities of GRL and UCM, the two complementary modeling notations of URN.

In the literature, it was stated that it was not necessary to use UCM or any business process modeling languages when creating models of the law. However, with the help of a simple example, we illustrated that there exist some situations such as with conditional statements which introduce the need for options and precedence. We can model these situations as a process using the UCM notation and create links between this view and GRL's. In future, we need to provide a more comprehensive case study to explore this idea and demonstrate the necessity of UCM. It is important to note that the UCM model of law does not need to cover all aspects of the law. We only turn to the capabilities of the UCM modeling language when some procedural implication is involved.

Modeling laws manually requires a much effort in indentifying legal element, their relationships and their interpretation in a goal model. In the future, it will become important to explore and support the automatic extraction of goal models, even if only partial, from legal documents.

Acknowledgments

This work was supported by a Collaborative Health Research Project grant from CIHR and NSERC (Canada) on *Performance Management at the Point of Care: Secure Data Delivery to Drive Clinical Decision Making Processes for Hospital Quality Control*.

References

1. Breaux, T.D., Vail, M.V., Antón, A.I.: Towards Regulatory Compliance: Extracting Rights and Obligations to Align Requirements with Regulations. In: 14th IEEE RE Conference, USA, pp. 49–58. IEEE CS, Los Alamitos (2006)
2. Breuker, J., Valente, A., Winkels, R.: Legal ontologies in knowledge engineering and information management. *Artificial Intelligence and Law* 12(4), 241–277 (2004)
3. Dardenne, A., van Lamsweerde, A., Fickas, S.: Goal-directed requirements acquisition. *Science of Computer Programming* 20(1-2), 3–50 (1993)
4. Darimont, R., Lemoine, M.: Goal-oriented analysis of regulations. In: REMO 2V 2006: Int. Workshop on Regulations Modelling and their Verification & Validation, June, Luxembourg (2006)

5. Delor, E., Darimont, R., Rifaut, A.: Software quality starts with the modelling of goal-oriented requirements. In: 16th International Conference Software & Systems Engineering and their Applications, Paris, France (December 2003)
6. Fuxman, A., Liu, L., Pistore, M., Roveri, M., Mylopoulos, J.: Specifying and Analyzing Early Requirements: Some Experimental Results. In: 11th IEEE International Requirements Engineering Conference, September 1993, pp. 105–114 (1993)
7. Ghanavati, S., Amyot, D., Peyton, L.: Towards a Framework for Tracking Legal Compliance in Healthcare. In: Krogstie, J., Opdahl, A.L., Sindre, G. (eds.) CAiSE 2007 and WES 2007. LNCS, vol. 4495, pp. 218–232. Springer, Heidelberg (2007)
8. Ghanavati, S., Amyot, D., Peyton, L.: A Requirements Management Framework for Privacy Compliance. In: Proceeding of the 10th Workshop on Requirements Engineering (WER 2007), Toronto, Canada, May, pp. 149–159 (2007)
9. Giorgini, P., Kolp, M., Mylopoulos, J.: Organizational patterns for early requirements analysis. In: Eder, J., Missikoff, M. (eds.) CAiSE 2003. LNCS, vol. 2681, pp. 617–632. Springer, Heidelberg (2003)
10. Giorgini, P., Mylopoulos, J., Nicchiarelli, E., Sebastiani, R.: Reasoning with goal models. In: Spaccapietra, S., March, S.T., Kambayashi, Y. (eds.) ER 2002. LNCS, vol. 2503, pp. 167–181. Springer, Heidelberg (2002)
11. Giorgini, P., Massacci, F., Mylopoulos, J., Zannone, N.: Requirements engineering meets trust management: Model, methodology, and reasoning. In: Jensen, C., Poslad, S., Dimitrakos, T. (eds.) iTrust 2004. LNCS, vol. 2995, pp. 176–190. Springer, Heidelberg (2004)
12. HIPAA, The Overview, <http://www.cms.hhs.gov/hipaaGenInfo> (accessed, January 2009)
13. ITU-T: User Requirements Notation (URN) – Language definition. ITU-T Recommendation Z.151 (11/08), Geneva, Switzerland (November 2008)
14. Rolland, C.: Reasoning with goals to engineer requirements. In: 5th International Conference on Enterprise Information Systems, Angers, France (April 2003)
15. Siena, A., Bonetti, A., Giorgini, P.: Balanced Goalcards: Combining Balanced Scorecards and Goal Analysis. In: Proceedings of the Third International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE 2008), Funchal, Portugal (May 2008)
16. Siena, A., Maiden, N.A.M., Lockerbie, J., Karlsen, K., Perini, A., Susi, A.: Exploring the effectiveness of normative i* modelling: Results from a case study on food chain traceability. In: Bellahsène, Z., Léonard, M. (eds.) CAiSE 2008. LNCS, vol. 5074, pp. 182–196. Springer, Heidelberg (2008)

A Requirement Engineering Framework for Electronic Data Sharing of Health Care Data Between Organizations

Xia Liu, Liam Peyton, and Craig Kuziemsky

University of Ottawa
550 Cumberland St. Ottawa, Ontario, Canada
xliu044@uottawa.ca, lpeyton@site.uottawa.ca,
kuziemsky@telfer.uottawa.ca

Abstract. Health care is increasingly provided to citizens by a network of collaboration that includes multiple providers and locations. Typically, that collaboration is on an ad-hoc basis via phone calls, faxes, and paper based documentation. Internet and wireless technologies provide an opportunity to improve this situation via electronic data sharing. These new technologies make possible new ways of working and collaboration but it can be difficult for health care organizations to understand how to use the new technologies while still ensuring that their policies and objectives are being met. It is also important to have a systematic approach to validate that e-health processes deliver the performance improvements that are expected. Using a case study of a palliative care patient receiving home care from a team of collaborating health organizations, we introduce a framework based on requirements engineering. Key concerns and objectives are identified and modeled (privacy, security, quality of care, and timeliness of service). And, then, proposed business processes which use new technologies are modeled in terms of these concerns and objectives to assess their impact and ensure that electronic data sharing is well regulated.

Keywords: Requirements Engineering, User Requirements Notation, health care, data sharing, privacy, quality of care.

1 Introduction

Currently, in spite of available information technology, health care providers still collect and share patient's information in an ad-hoc basis, by paper-based forms, faxes and phone. Many of the reasons for this are related to privacy and security concerns, but there is also a resistance to technology and uncertainty that investment in technology will actually result in cost-effective improvements to healthcare. The potential convenience of electronic healthcare data sharing is often overlooked. In the group of health care providers, nurses are the major persons who are responsible for collecting patient's data and entering into the electronic system. One study of information technology for palliative care showed that nurses were required to do "double entry" into electronic systems and paper charts for various reasons including medical legal issues [1]. Such duplicate work creates the opportunity for medical errors and

makes care providers (physicians, nurses etc.) hesitate to accept electronic records. Much of this duplicate work is motivated by studies showing electronic healthcare delivery to be problematic [2] which raises questions about the extent it can enhance care delivery. It is important in any switch to new technology to be able to monitor and document that quality of care is being maintained.

This paper describes a requirements management framework for electronic health care data sharing that will help healthcare providers model and evaluate new and existing healthcare processes in order to validate alignment with quality of care goals and policies as well as document compliance. Based on a case study of palliative care, we develop the framework using an ITU standard notation: User Requirements Notation (URN). In the framework, we show how to model the improvements obtained by electronic data sharing, but also address concerns like privacy, security and quality of care.

2 Background

It is expected that the number of individuals suffering from and living with chronic illness such as diabetes, heart disease and cancer will increase significantly in the forthcoming years. Providing care for chronic illness requires a movement from care delivery by a single provider and location to care delivery by multiple providers across multiple settings. Team based care delivery is challenging for the fundamental reason that our healthcare system is not designed to deliver such care. The electronic health record (EHR) provides the means for electronic data collection but there is still a need to support the underlying care delivery processes that take place. Information access and sharing must be timely, accurate and secure or quality of care delivery can suffer. Poor information sharing in team based care delivery can be a source of medical errors [3]. Thus if we are to support team base care delivery we must facilitate data sharing but also support and monitor the underlying business processes that use the data. Stead et al. [4] point to the need for an informatics infrastructure that details how to link information and business process needs to enable us to design technological solutions that provide care when and where needed, supporting processes that avoid error, and provide quality care while reducing administrative costs. The framework presented in this paper provides the basis for such an informatics infrastructure to support team based care delivery.

Our case study is taken from a health care jurisdiction in Ontario, Canada, where the applicable privacy legislation is the Personal Health Information Privacy Act [5]. PHIPA specifies the legal responsibilities of health information custodians in terms of how they are to handle personal health information. PHIPA is legislation specific to healthcare in the Canadian province of Ontario within the framework of the federal Personal Information Protection and Electronic Documents (PIPEDA) act [6]. PIPEDA has been recognized by the European Commission as being compliant with the European Union's Data Protection [7]. In the United States, there is similar legislation for healthcare in the form of the Health Insurance Portability and Accountability Act (HIPAA) [8].

Researchers have worked on applying requirement engineering concepts and tools to provide methodologies to ensure compliance and traceability between organizational

goals and the business process that are supposed to achieve those goals. [9] describes how to apply one of the main goal-oriented requirements engineering methodologies (KAOS) to model regulations. They explain how to transform regulation documents into goal, objects and threat models incrementally and how to maintain a level of traceability from the source document to those models. [10] introduced the Requirement-based Access Control Analysis and Policy Specification (ReCAPS) method to integrate access control analysis, improve software quality and develop policy and requirements-compliant systems. This method emphasizes compliance between different policy levels, requirements and system designs. In [11], *i**, a modeling language similar to Goal-oriented Requirement Language (GRL), was used to design information systems within a social context. In [12], User Requirements Notation (URN) was used as a basis for a framework to track legal compliance between health care processes and privacy legislation.

We will use URN as a basis for our framework as well. URN was designed for modeling and analyzing requirements in the form of goals and scenarios prior to design [13]. It can be used to model most kinds of reactive and distributed systems, as well as business processes [14]. URN combines two complementary notations: the Goal-oriented Requirement Language (GRL) and Use Case Maps (UCM) which are used for modeling goals and processes respectively. Figure 1 and figure 2 show a brief summary of these two notations.

URN is a draft ITU-T standard [13] that combines goals and scenarios in order to help capture, model and analyze user requirements at the early stages of design. It can be applied to describe most kinds of reactive and distributed systems as well as business processes. URN is the only modeling language that can model goals and processes at the same time while providing traceability between them. URN integrates two notations, namely the Goal-oriented Requirement Language (GRL) and Use Case Maps (UCM). GRL is used to model, with AND/OR graphs, the relationships and strategies around how actors and tasks are organized to achieve goals and objectives. Figure 1 shows a subset of the main GRL elements. An example of a GRL diagram is shown in the figure 3.

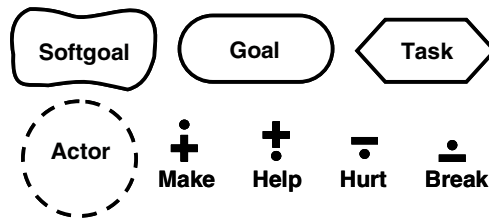


Fig. 1. Subset of GRL notation

The UCM notation is used to model business processes and system behaviour in terms of related scenarios and use cases. Scenario paths connect start points (pre-conditions and triggering events), end points (post-conditions and resulting events), and responsibilities. Responsibilities indicate where actions, transformations, or processing are required. They can be performed in sequence, concurrently (using AND-forks and AND-joins) or as alternatives (with guarded OR-fork and OR-join). Complex

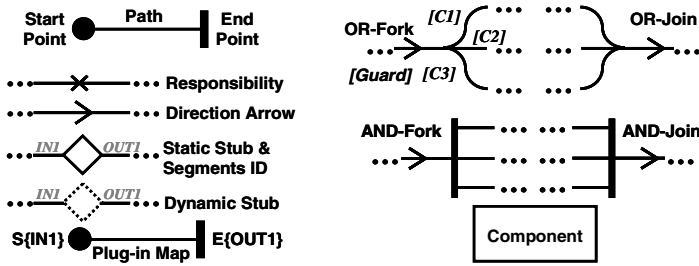


Fig. 2. Subset of UCM notation

processes can be defined at any level of abstraction and be decomposed with stubs, which act as containers for sub-maps. The subset of the UCM notation used in this chapter is shown in Figure 2, and an example of a UCM diagram is shown in Figure 4.

The UCM process view specifies the responsibilities to be performed (the *what* aspects) by *whom*, *when*, and *where*. The GRL goal view provides a rationale (*why*) for the business process elements, together with an explanation of why alternative solutions were chosen or not. More details on URN are provided in [15] [13]. A detailed analysis of the capabilities of URN in comparison with other well-known business process modeling languages is given in [16].

URN models are built using the Eclipse-based jUCMNav tool [17]. jUCMNav supports an extensible meta-model for extending the set of diagrams, model elements and links the tool can work with as well as a data exchange layer for integration with other tools and systems. [18]

3 Palliative Care Scenario

Palliative care is care provided to patients at end of life when curative therapies are not an option. Palliative care is an ideal domain to study team based care delivery across multiple settings as that is an integral part of palliative care delivery. [19]

In this scenario, the health authority responsible for palliative care in a region of Ontario, Canada proposes to build a palliative care information system (PAL-IS). The intent is that PAL-IS will facilitate sharing of patient information among healthcare providers such as doctors, nurses and case managers as well as support the underlying processes of care delivery such as decision making and treatment dissemination. This goal for PAL-IS is consistent with the overall goal of palliative care, which is to improve the life quality for patients who have life-threatening illness and their families.

Improving patients’ access to healthcare services and delivery of healthcare services from providers are also important goals for PAL-IS. Technically, PAL-IS should ensure communication between patients and healthcare providers is efficient and timely. PAL-IS should support and ensure patients can get timely access to their nurses or physicians whether in hospital, home or other care centers and healthcare providers can respond to patients quickly based on their requests or needs. If care issues can be identified and managed efficiently then their hospital stays can be shortened or avoided.

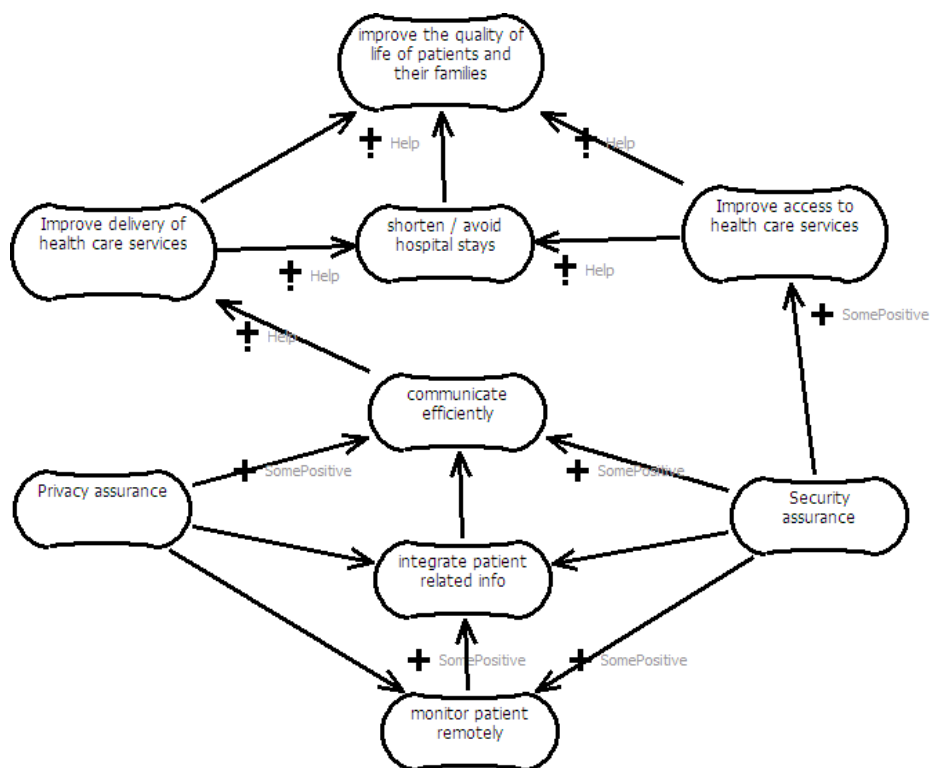


Fig. 3. The goals of PAL-IS

The most important challenge is to establish a framework and process for ensuring that PAL-IS meets the needs of the region for palliative care. In such an information system, patients are being monitored remotely and their personal health information is being integrated among different organizations. Privacy and security concerns arise.

The goal model for the health care organization and PAL-IS is described as a GRL model in figure 3. Improving the life quality of patients and their families is the top goal in the health care service, and monitoring patients remotely together with ensuring privacy and security at the same time is one of the basic objectives our system should meet.

The goals in Figure 3 are not completely independent; they affect or contribute to each other. Privacy and security concerns, as the most basic goals, should be fulfilled first. Actually, without privacy and security assurance, monitoring patients remotely, integrating patient information and communication among organizations are not allowed to proceed, even if there is no technology problem at all.

With assurance of privacy and security, PAL-IS can provide timely access to palliative care services for patients or their families from home, and promote the services delivered to their home quickly, and finally reach the top goal of improving the quality of life of patients and their families.

In figure 4, we use a UCM diagram to document a key business process or scenario that PAL-IS must support for pain management. A cancer patient is on two medications for his pain. A homecare nurse and a physician are monitoring the patient’s symptoms through PAL-IS. One of the patient’s daily jobs is to send his pain score to the nurse through the system. There are four pain alerts with different priorities in the system, depending on the pain scores the patient sent. If the patient enters a low number for the pain score, the alert is set at a low priority. The number will be recorded and the nurse will simply continue monitoring. But if the patient enters a high number for the pain score indicating severe pain, the alert would be set as a high priority and the nurse would contact the physician for appropriate action such as getting an updated prescription. Once a new prescription is issued, the nurse will send it back to the patient. [20]

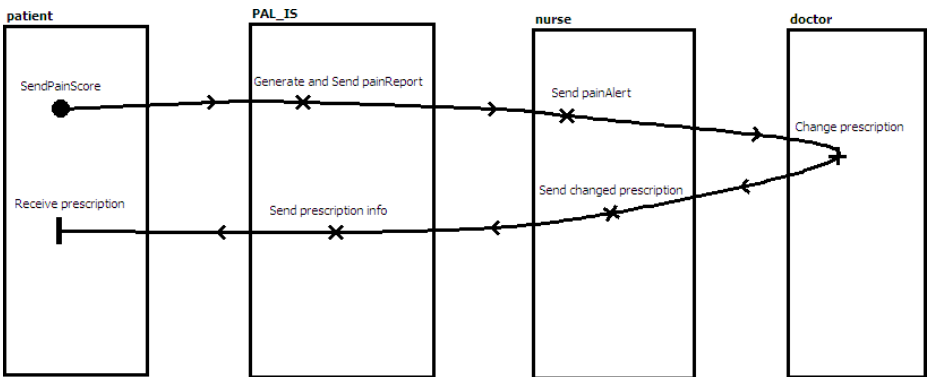


Fig. 4. PainScore reporting and PainAlert generating processes in PAL-IS

For comparison, figure 5 shows the current process that must be followed without the PAL-IS system. The nurse records the patient’s pain score manually and periodically and must visit the patient in person, or contact them over the phone. Based on the paper version records, the nurse is responsible for analyzing the pain scores, generating a pain score report and making a judgment if a pain alert is a low priority or a high one. The nurse is also the person to send the pain alert to the corresponding doctor, again by calling or faxing a paper-based document. If the doctor receives the pain alert document with a high priority, they will write a new prescription and send it back the nurse, who is going to send it to the waiting patient. All the steps are manual and time-consuming, especially for the nurse to analyze pain scores and generate a pain score report.

In the above manual process, there are a few measurements to make sure the patient is properly taken care:

- The nurse should talk to the patient and take his/her pain score at least once every 4 hours.
- The nurse should analyze the patient’s pain scores and update his/her pain score report at least once every 4 hours.
- Once a pain alert with a high priority is issued, the patient should receive his/her changed prescription within one hour.

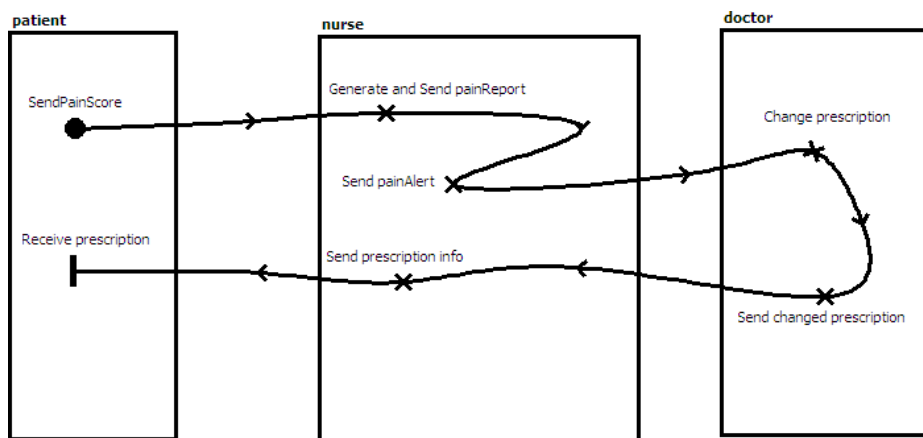


Fig. 5. PainScore reporting and PainAlert generating processes without PAL-IS

Since all the steps are manual, the nurse, as the main responsible person in the document generating and exchanging process, is overloaded. Now, let us go back to figure 4, the painScore reporting and prescription changing process in the PAL-IS. With the help of an Information System, the nurse will not have to manually take records from the patient and generate the pain score report. The following are the steps in this process:

1. The patient will enter his/her pain score periodically into the system.
2. The information system will update the patient's pain score report automatically and give a pain alert number indicating a low or high priority. The pain score report and the pain alert are sent to the nurse's computer.
3. The nurse will process the pain alerts based on priorities, and send them electronically to corresponding doctors.
4. For a pain alert with high priority, the doctor will issue a new prescription and send back the nurse electronically.
5. The changed prescription is finally sent from the nurse to the patient through the information system.

From the above example, we see that the process sharing healthcare data through PAL-IS is completely electronic. Also, there are some measurements to improve care, which are different from those in the manual process:

- To use PAL-IS, all users need be identified for security purposes.
- The patient should enter a pain score at least once every 4 hours.
- If the information system did not receive the patient's two consecutive pain scores, a low priority alert would be raised and sent to the nurse. The nurse should contact the patient immediately by phone.
- If the patient entered a pain score greater than 7/10, a pain alert should be issued immediately from the information system to the nurse and then to the doctor. The patient should receive the changed prescription within one hour.

4 Framework

Figure 6 depicts the processes involved in the electronic data sharing of healthcare data between organizations, based on the PAL-IS case study. In this model, there is a central information system, through which organizations send and receive patients' data, to reach the objective of electronic healthcare data sharing.

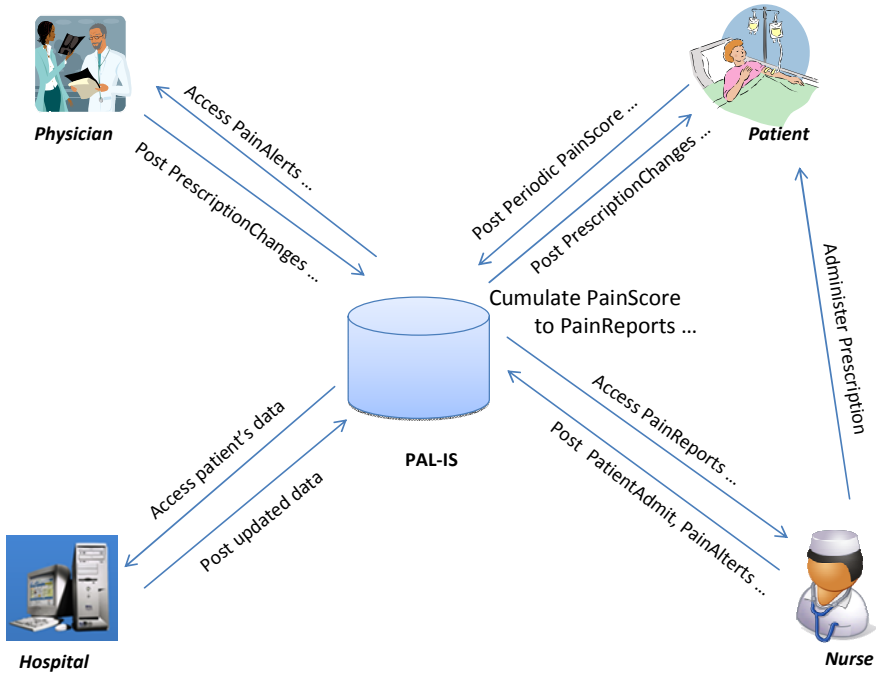


Fig. 6. Model of Electronic Data Sharing between Organizations

In addition to enabling the electronic flow of information between patient, nurse and doctor, the information system is also able to collect and report statistics on how efficiently and effectively care is being provided. The GRL diagram can be amended to indicate tasks (linked to the appropriate UCM diagram for detailed analysis) which are critical to the goals of the organization. Associated with those tasks, can be metrics which measure how effectively and efficiently care can be provided. This is depicted below in figure 7.

Figure 7 shows how the processes involved in PAL-IS affect the goals defined in figure 3. Here, we provide three processes as examples. A patient is supposed to enter a pain score at least once every 4 hours. A pain score report is created, based on the most recent pain scores from the patient, and sent to a nurse for review. If the patient failed to enter two consecutive pain scores or if a pain score greater than 7/10 was entered, the information system will raise a pain alert.

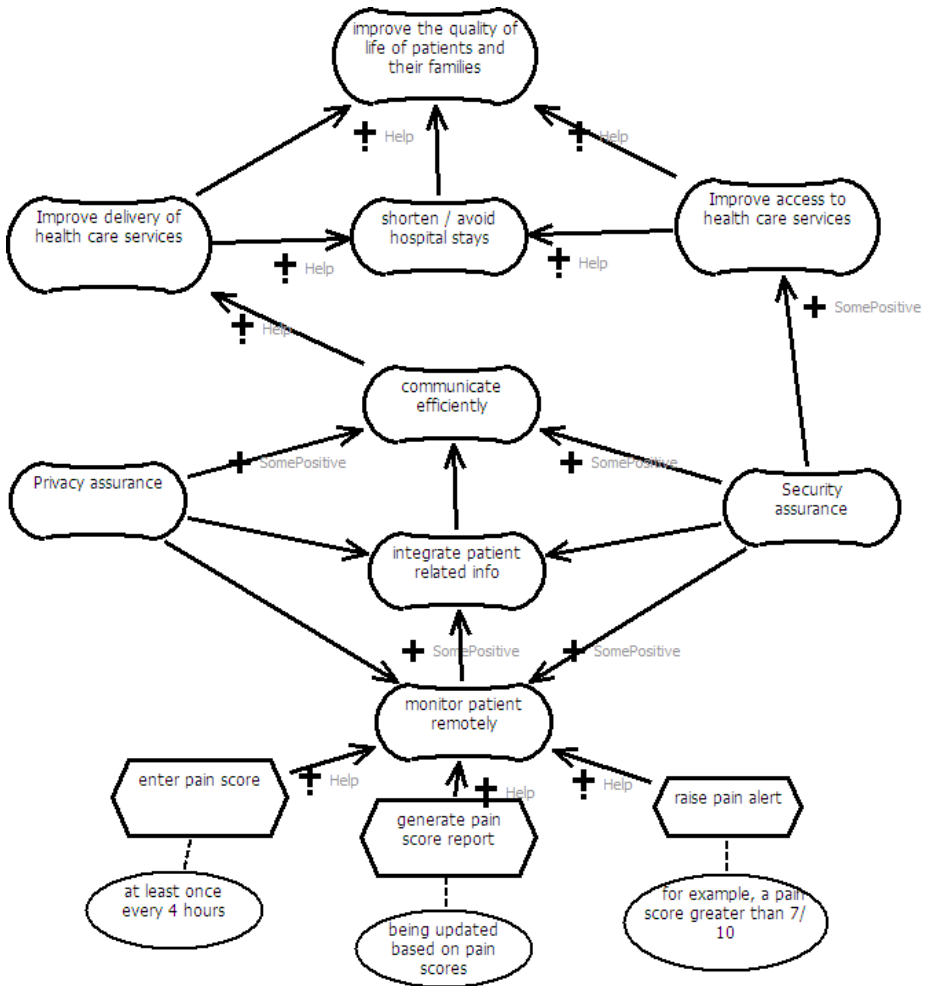


Fig. 7. The GRL diagram with linked tasks

5 Conclusions

In evaluating our framework, we can see that in using URN, the healthcare provider is able to not only articulate their goals, but link them to the relevant business processes. In the case where the business processes are supported by an information system, they can further define and mandate additional data collection and reporting on metrics to monitor how effectively the organization goals are being achieved. In order to achieve this, though, the healthcare provider will need to invest in modeling explicitly both the goals the provider is trying to achieve and the processes that are enacted to realize them, as well as identifying appropriate metrics to measure progress, and ensure that the necessary data is collected.

Acknowledgements

This work was supported by a Collaborative Health Research Project grant from CIHR and NSERC (Canada) on Performance Management at the Point of Care: Secure Data Delivery to Drive Clinical Decision Making Processes for Hospital Quality Control.

References

- [1] Kuziemsky, C.: Information Technology in Palliative Care, Working Paper, Action for Health Project. University of Victoria (July 2004), <http://www.sfu.ca/act4hlth/pub/working/IT%20Palliative.pdf> (accessed, February 2009)
- [2] Ash, J.S., Berg, M., Coiera, E.: Some unintended consequences of information technology in health care: the nature of patient care information system-related errors. *J. Am. Med. Inform. Assoc.* 11(2), 104–112 (2004)
- [3] Alvarez, G., Coiera, E.: Interdisciplinary communication: An uncharted source of medical error? *Journal of Critical Care* 21, 236–242 (2006)
- [4] Stead, W.W., Kelly, B.J., Kolodnder, R.M.: Achievable Steps Toward Building a National Health Information Infrastructure in the United States. *J. Am. Med. Inform. Assoc.* 12, 113–120 (2005)
- [5] PHIPA, Government of Ontario: Personal Health Information Protection Act (2004), http://www.e-laws.gov.on.ca/html/statutes/english/elaws_statutes_04p03_e.htm (accessed, January 2009)
- [6] PIPEDA, Government of Canada, Health Information Custodians in the Province of Ontario Exemption Order (2005), <http://canadagazette.gc.ca/partII/2005/20051214/html/sor399-e.html> (accessed January, 2009)
- [7] European Union, Directive on Privacy and Electronic Communications. European Parliament, Brussels, Belgium, (2002), http://europa.eu/eur-lex/pri/en/oj/dat/2002/1_201/1_20120020731en00370047.pdf (accessed, January 2009)
- [8] HIPPA, United States Department of Health and Human Services, Medical Privacy - National Standards to Protect the Privacy of Personal Health Information (1996), <http://aspe.hhs.gov/admsimp/pl1104191.htm> (accessed, January 2009)
- [9] Darimont, R., Lemoine, M.: Goal-oriented Analysis of Regulations. In: International Workshop on Regulations Modelling and their Verification & Validation (REMO2V 2006). Presses Universitaires de Namur, Luxemburg (2006)
- [10] He, Q., Otto, P., Ant'ón, A.I., Jones, L.: Ensuring compliance between policies, requirements and software design: A case study. In: IWIA 2006: Proc. Fourth IEEE Int. Workshop on Information Assurance, Washington, USA, pp. 79–92. IEEE Computer Society, Los Alamitos (2006)
- [11] Liu, L., Yu, E.: Designing Information Systems in Social Context: A Goal and Scenario Modelling Approach. *Info. Systems* 29(2), 187–203 (2004)
- [12] Ghanavati, S., Amyot, D., Peyton, L.: A Framework for Tracking Legal Compliance in Health Care. In: Krogstie, J., Opdahl, A.L., Sindre, G. (eds.) CAiSE 2007 and WES 2007. LNCS, vol. 4495, pp. 218–232. Springer, Heidelberg (2007)

- [13] ITU-T, Recommendation Z.150 (02/03): User Requirements Notation (URN) – Language requirements and framework, Geneva, Switzerland, 200337
- [14] Weiss, M., Amyot, D.: Business Process Modeling with URN. *International Journal of E-Business Research* 1(3), 63–90 (2006)
- [15] Amyot, D.: Introduction to the User Requirements Notation: Learning by Example. *Computer Networks* 42(3), 285–301 (2003)
- [16] Mussbacher, G.: Evolving Use Case Maps as a Scenario and Workflow Description Language. In: 10th Workshop of Requirement Engineering (WER 2007), Toronto, Canada, May 2007, pp. 56–67 (2007)
- [17] Roy, J.-F., Kealey, J., Amyot, D.: Towards Integrated Tool Support for the User Requirements Notation. In: Gotzhein, R., Reed, R. (eds.) SAM 2006. LNCS, vol. 4320, pp. 198–215. Springer, Heidelberg (2006)
- [18] Kealey, J., Kim, Y., Amyot, D., Mussbacher, G.: Integrating an Eclipse-Based Scenario Modeling Environment with a Requirements Management System. In: 2006 IEEE Canadian Conf. on Electrical and Computer Engineering (CCECE 2006), Ottawa, Canada, pp. 2432–2435 (2006)
- [19] Cummings, I.: The interdisciplinary team. In: Dovle, D., Hanks, C.W.C., MacDonald, N. (eds.) *Oxford Textbook of Palliative Medicine*, 2nd edn., pp. 19–30. Oxford University Press, Oxford (1998)
- [20] Kuziemyky, C.: *Palliative Healthcare Patient Scenario v1.0* - July 2008 (2008)

An Aspect-Oriented Framework for Business Process Improvement

Alireza Pourshahid¹, Gunter Mussbacher¹, Daniel Amyot¹, and Michael Weiss²

¹ SITE, University of Ottawa, 800 King Edward Ave
Ottawa ON, K1N 6N5 Canada

² SCE, Carleton University, 1125 Colonel By Drive, Ottawa, ON, K1S 5B6 Canada
apour024@uottawa.ca, gunterm@site.uottawa.ca,
damyot@site.uottawa.ca, weiss@sce.carleton.ca

Abstract. Recently, many organizations invested in Business Process Management Systems (BPMSs) in order to automate and monitor their processes. Business Activity Monitoring is one of the essential modules of a BPMS as it provides the core monitoring capabilities. Although the natural step after process monitoring is process improvement, most of the existing systems do not provide the means to help users with the improvement step. In this paper, we address this issue by proposing an aspect-oriented framework that allows the impact of changes to business processes to be explored with what-if scenarios based on the most appropriate process redesign patterns among several possibilities. As the four cornerstones of a BPMS are process, goal, performance and validation views, these views need to be aligned automatically by any approach that intends to support automated improvement of business processes. Our framework therefore provides means to reflect process changes also in the other views of the business process. A health care case study presented as a proof of concept suggests that this novel approach is feasible.

Keywords: Business Process Management, Aspect-Oriented Modeling, User Requirements Notation, Process Improvement, Process Redesign, Process Modeling.

1 Introduction

Business Process Management (BPM) has recently gained momentum among e-business technologies. BPM can be realized through methodologies, techniques, or software, in a way that helps organizations bring together processes and their context including people, documents, information sources, organizational structures, and applications [1]. As a methodology, BPM helps organizations gain control over their business processes by modeling, validating, analyzing, and monitoring the processes. BPM provides process visibility for the organizations, and hence makes both human-centric and electronic-centric processes more manageable [2].

A BPM methodology is typically an iterative lifecycle composed of several steps, which usually starts with modeling and validation of the business processes. The next steps in the lifecycle are the automation and execution of these processes. Then, the

processes are monitored and finally, based on the monitoring results, they may be redesigned and improved to better achieve the expected goals [3]. In addition to such methodologies, Business Process Analysis (BPA) ontologies have been suggested to make the analysis effort more effective and to reduce the gap between IT and the business world [4]. Furthermore, there has been some work done to capture common improvement approaches in the form of business redesign patterns that contribute to the improvement of processes from four main perspectives, namely time, quality, cost, and flexibility [5]. Although these redesign patterns could be used as a guideline for the improvement step, most of the improvement methods do not utilize these patterns. On the contrary, these methods rely heavily on human innovation and creativity rather than on rationality. Using patterns can help to further rationalize and formalize process improvement methods [6].

BPM as software is often called a Business Process Management System (BPMS). Existing BPMSs, such as Appian enterprise BPM suite, G360, Tibco iProcess Suite, EMC BPM Suite, and Fujitsu Interstage BPM Suite [7], provide various methods for process monitoring. Monitoring in these systems is usually done by defining calculation rules for specific measurement points which quantify important business concerns called Key Performance Indicators (KPIs) [8]. However, these systems usually do not provide the means for process improvement. In addition, they do not support the application of process redesign patterns. Therefore, improvements can be done solely based on human knowledge and experience [9].

Furthermore, available process modeling notations such as BPMN, UML, EPC, YAWL, and IDEF3 do not provide means to observe or simulate the impact of such patterns on KPIs and business goals before the patterns are implemented in the system [10, 11]. They also do not allow the comparison of candidate redesign patterns with respect to their impact on high-level business goals. Therefore, support for effectively selecting the most appropriate pattern for the current business situation is very limited. Essentially, these features are not supported because there is no method to automatically reflect changes to the process model in the KPIs and business goal models – possibly even based on historical expectations. Consequently, a change to the process model requires all the other related models to be tracked and modified manually, which can be a tedious and error-prone task.

We are proposing an aspect-oriented framework based on the User Requirements Notation (URN) standard [12] to address the aforementioned issues. URN is an integrated process/goal modeling notation that was recently extended for business process monitoring [13]. Four complementary views for a business problem can be defined with the help of URN, including a process view, a goal view, a validation view, and a performance view [3]. We can use these views to monitor a process and identify process deficiencies. Aspect-oriented URN (AoURN) [14], on the other hand, extends URN with aspect-oriented concepts and can be used to define redesign patterns in a modular way, as aspects. Aspects can be merged with the existing process model to explore the impact of applying a redesign pattern on business goals, and such changes can be easily undone if the desired process improvements do not materialize, thus lowering the barrier to apply redesign patterns. Technically, AoURN's pointcut expressions define criteria that describe the parts of the process to which the redesign patterns should be applied. AoURN also provides the capability to define corresponding aspects for each of the pointcut expressions and apply them to the process model.

Applying the aspect is equivalent to applying the redesign pattern, thus modifying the process and replacing the appropriate parts with a new model. Moreover, AoURN provides similar functionality for the goal and performance views. Using such an approach, one can apply several process redesign patterns to a process, align the goal/performance views accordingly, and compare the results to select the best overall pattern based on the impact of redesign patterns on overall business goals.

This paper contributes to the body of knowledge in several ways. First, we provide an innovative method for selecting the most appropriate redesign pattern among several candidate patterns, considering the impact of the patterns on process performance and business goals. Second, we further enhance the existing URN based process monitoring framework with a repository of redesign patterns that supports the exploration of proposed changes with what-if scenarios based on these patterns. Third, our suggested approach allows for dynamic and continuous adaptation of business processes to current business needs based on the defined KPIs. Finally, this work introduces the first example of AoURN pointcut expressions that span goal and process models. This allows us to not only define related aspects on goal and process views but also to apply them to both views at the same time based on criteria formulated not just for one view but for both types of views. Validation of the framework is done in part through a health care case study.

Although our approach currently concentrates on the capabilities of URN to elaborate our novel concepts and illustrate them, the ideas presented here could be used in other contexts and as a generalized methodology.

The rest of this paper is organized in four main parts. In section 2, we elaborate on the background knowledge required to understand the new concepts introduced in this paper including BPM using URN as well as AoURN. In section 3, we describe the new aspect-oriented framework for business process improvement. In section 4, we illustrate the application of this framework in a health care case study. Finally, the last section discusses our conclusions and future work.

2 Background and Related Work

URN-based Business Process Management, process redesign patterns, and the Aspect-oriented User Requirements Notation provide the basis for the framework introduced in this paper, and we elaborate on these concepts in sections 2.1 and 2.2.

2.1 BPM with URN

The User Requirements Notation (URN) is an International Telecommunication Union standard for capturing early requirements in the form of scenarios and goals [12, 15]. URN consists of two complementary sub-languages called Goal-oriented Requirement Language (GRL) and Use Case Maps (UCM) for goal modeling and scenario modeling, respectively [15]. In this paper, we will introduce basic notation elements as we go.

Fig. 1 and Fig. 2 illustrate a hospital's Data Warehouse Approval Process used as a case study in this paper. Fig. 1 is a GRL model consisting of GRL intentional elements linked together using contribution links. Elements are called intentional because they

carry stakeholder intentions. Contribution links indicate the impact of intentional elements on each other – in this case, positive qualitative contributions are shown. Three types of intentional elements have been used in this model. Soft goals (◻, e.g., Reduce Costs) describe something to be achieved that cannot be measured quantitatively but is of a qualitative nature, tasks (◻, e.g., Approval Process) model potential solutions for achieving higher-level goals, and KPIs (◻, e.g., Number of Mistakes) indicate metrics of the system normalized to a scale of -100 to 100.

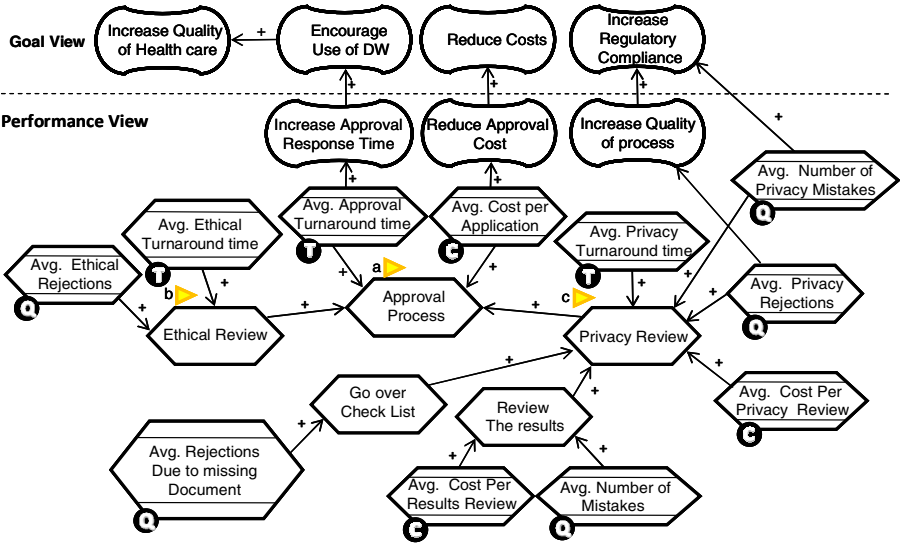


Fig. 1. Data Warehouse Approval Process – Goal and Performance View

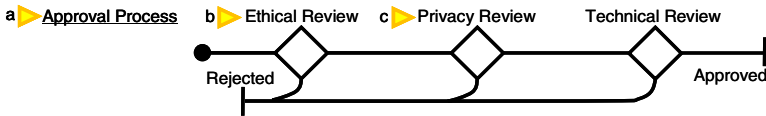


Fig. 2. Data Warehouse Approval Process – Process View

In addition, URN traceability links (▶) are used to relate tasks in this goal model to their representation in the UCM model, Fig. 2. The Approval Process is linked to the whole diagram, while the other two tasks are linked to individual model elements. The UCM model provides further behavioral details about the linked tasks. It consists of a path that begins at a start point (●) and ends with an end point (▬). Stubs (◊, e.g., Ethical Review) are containers for sub-models and denote here the three major steps in the process. Drilling into a stub leads to a submap which provides more details about the step. Note that in this example, each stub has two possible exits: one if a review approved the data warehouse access request and one if it was rejected.

Business process modeling using URN was introduced in [16] and [17] by modeling, analyzing, and evolving a supply chain management system. While the current

version of the URN notation still requires better support for exception and cancellation handling in process and workflow models [18], its unique capabilities for modeling both processes with UCMs and goals with GRL in a unified way is a significant advantage over other process modeling notations [10]. The integrated view of UCM and GRL not only answers the *where*, *what*, *who*, and *when* questions of process models, but also answers *why* a particular part of a process exists. Using URN, people with sufficient business knowledge and experience can align business goals and processes [19].

Business process modeling using URN was introduced in [16] and [17] by modeling, analyzing, and evolving a supply chain management system. While the current version of the URN notation still requires better support for exception and cancellation handling in process and workflow models [18], its unique capabilities for modeling both processes with UCMs and goals with GRL in a unified way is a significant advantage over other process modeling notations [10]. The integrated view of UCM and GRL not only answers the *where*, *what*, *who*, and *when* questions of process models, but also answers *why* a particular part of a process exists. Using URN, people with sufficient business knowledge and experience can align business goals and processes [19].

jUCMNav [20] is the most comprehensive URN modeling tool available today. By formalizing the data exchange layer [21], external information systems such as data warehouses can be connected to the *jUCMNav* tool. Integration with external tools including a Requirement Management System (RMS), DOORS [22], and a Business Intelligence System (BIS), Cognos [9], as well as extending URN [13] helped with the development of an integrated BPM framework [3] for process validation [23] and process performance monitoring [11].

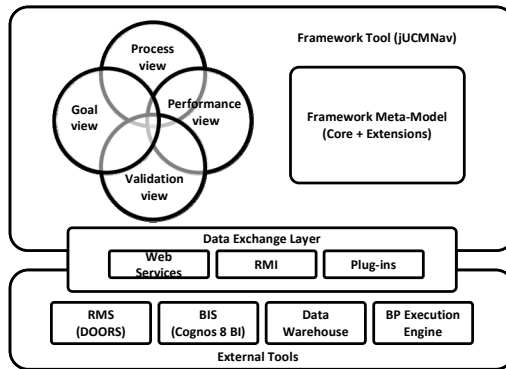


Fig. 3. Framework Core Views and Components

The developed BPM framework consists of four main views and several components (see Fig. 3). The process view captures the workflow and the sequence of steps in the business process, from very high levels of abstraction down to the task and responsibility level describing the atomic parts of the process. The goal view captures the business goals related to the process. Goal modeling can focus initially on high-level strategic

goals of the business, which are later decomposed into low-level operational goals and even tasks. It is common to see the same tasks shared in both the low-level goal view and the low-level process view. Therefore, the process and goal views can be associated together, defining which part of a process impacts which business goals. In addition, the performance view introduced in [11] is associated with both the process and goal views and illustrates how processes perform with respect to the business goals using Key Performance Indicators. Finally the validation view introduced in [24] defines the requirements and restrictions against which the process view should be validated. For instance, corporate policies, laws, or service level agreements can be considered as some of these validation criteria that need to be satisfied by the process view.

The BPM framework uses the built-in evaluation mechanisms of URN to evaluate high-level business goals and other validation criteria based on the satisfaction levels of low-level goals and the values of KPIs. For example, monitoring the Data Warehouse Approval Process (see Fig. 1) yields process measures that in turn result in initial satisfaction values for the KPIs. These values are then propagated to higher-level nodes in the goal graph until the highest-level goals have been evaluated.

The BPM framework is an iterative and incremental approach with several steps for business process improvement. In the first step, the target processes for improvement are selected. This selection can be done based on the priorities of the organization. Then, the artifacts required for the improvement, including the four views, are modeled and the association links between the views are established. Subsequently, the dimensional data sources used for monitoring are prepared and the performance of the processes is monitored. In the alignment step, the views are modified to address the issues observed from the monitoring step. The framework provides guidelines required for process improvement by suggesting the necessary artifacts and context information for rational analysis of the processes, thus identifying possible improvement points. Although, this framework has been designed to use the process redesign patterns in the alignment step, it does not provide the means for helping analysts to select the best design pattern according to the process status and organization goals. The selection of the pattern still requires the analysts to go through all the patterns and heavily relies on their knowledge and expertise.

In this paper, we move toward providing help for the analysts in terms of selecting and applying the design patterns. Generalized redesign patterns as suggested in [5] as well as customized alternatives appropriate for specific processes can be utilized. The generalized patterns discussed in [5] can help with process improvement in four categories including cost, time, quality, and flexibility. Table 1 shows the three redesign patterns used in this paper as examples. A complete list of the redesign patterns with their impact on the four categories is available in [13].

Table 1. Sample Redesign Patterns and their Impact on the Four Categories

Redesign Pattern	Time	Cost	Quality	Flexibility
Knockout	↓↑	↑	N/A	↓
Task Elimination	↑	↑	↓	N/A
Control Relocation	N/A	N/A	↑	N/A

↑: Positive Impact ↓: Negative Impact †: Maybe Positive Impact ‡: Maybe Negative Impact

2.2 Aspect-Oriented Modeling With the User Requirements Notation

The Aspect-oriented User Requirements Notation (AoURN) is a modeling framework that extends URN with aspect-oriented concepts [14], allowing modelers to better encapsulate crosscutting concerns which are hard or impossible to encapsulate only with URN models. With AoURN, business process redesign patterns can more easily be encapsulated as concerns in their own modules and selectively applied to the existing process. AoURN treats concerns as first-class modeling elements. AoURN groups all relevant properties of a concern such as goals, behavior, and structure, as well as pointcut expressions needed to apply new goal and scenario elements to a base model or to modify existing elements. A *pointcut expression* is a pattern that must be matched in the base model if the aspect is to be applied, thus determining the base model locations to which the aspect is applied.

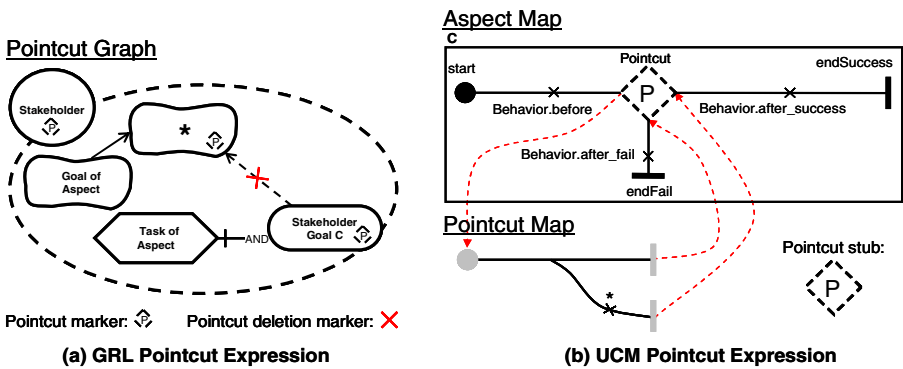


Fig. 4. GRL and UCM Pointcut Expressions

AoURN adds aspect concepts to URN’s sub-notations, leading to Aspect-oriented GRL (AoGRL) [25] and Aspect-oriented UCMs (AoUCM) [26, 27]. AoURN uses standard URN diagrams to describe pointcut expressions (i.e., it is only limited by the expressive power of URN itself as opposed to a particular composition language). GRL pointcut expressions are shown on a *pointcut graph* and make use of *pointcut (deletion) markers* to indicate the pattern to be matched (see Fig. 4.a). All elements without pointcut markers are added to the matched location in the GRL model, while elements with a pointcut deletion marker are removed. Goals and tasks of an aspect may be described in more detail in separate goal graphs called *aspect graphs*.

UCM pointcut expressions define the pattern to be matched with a *pointcut map* (see Fig. 4.b). The aspectual properties are shown on a separate *aspect map*. The aspect map is linked to the pointcut expression with the help of a *pointcut stub*. The causal relationship of the pointcut stub and the aspectual properties visually defines the composition rule for the aspect, indicating how the aspect is inserted in the base model (e.g., before, after, instead of, in parallel, interleaved, or anything else that can be expressed with the UCM notation).

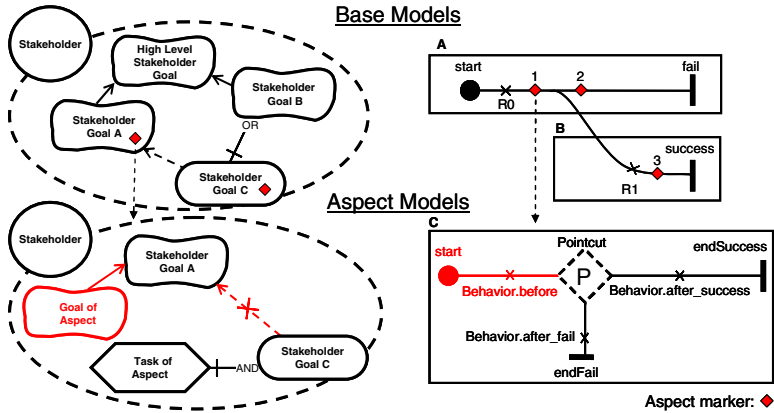


Fig. 5. Composed Model with Aspect Markers

AoURN employs an aspect composition technique that can fully transform URN models. Locations affected by an aspect are indicated by *aspect markers* (see Fig. 5). When an aspect marker is selected, the modeler is taken to an aspect view with only those aspectual properties highlighted that are relevant to the selected aspect marker.

3 Framework

The intention of the suggested framework is to improve processes that do not satisfy their goals by applying the most appropriate redesign pattern. Since this framework builds on the BPM framework introduced in section 2.1, one has to first define the three vital artifacts required for process monitoring (i.e., the process, goal, and performance views). Furthermore, the process redesign patterns must be modeled as aspects which are later applied to the existing process by defining pointcut expressions. Customized redesign patterns or alternative versions of the processes may be modeled by the users of the framework in addition to standard redesign patterns. To aid the selection of the appropriate redesign pattern, KPIs are categorized according to four redesign pattern groups (i.e., cost, time, quality, and flexibility). The filled circles with T, Q, and C in Fig. 1 indicate the time, quality, and cost categories for the KPIs, respectively.

As part of the performance modeling step, each KPI is normalized. The worst possible value, a threshold value, and the target value are defined for real world values of each KPI and then mapped to -100, 0, and 100 on the GRL scale, respectively. Now, the business processes may be monitored. Any KPI value that is not satisfactory (i.e., that is far off the target value) indicates a possible area of improvement. The KPI categories help determine the candidate redesign patterns. For instance, if the KPI with the unsatisfactory value is categorized as a time KPI, all redesign patterns with a positive impact on time are possible candidates for improving the observed values of the KPI and consequently the process. However, all of the possible candidate redesign patterns may not be applicable to the target process for which the KPI has been

defined. Therefore, in the next step, we use additional characteristics of redesign patterns to reduce the number of candidates. These characteristics are identifying features of the redesign pattern that are expressed with AoURN pointcut expressions, e.g., the redesign pattern may require a certain sequence of process model elements or may require that certain KPIs perform worse than other KPIs. A redesign pattern, therefore, is only applicable to the process model, if a match of its pointcut expression can be found in the process model. After identifying the applicable patterns and if more than one possible choice exists, users can apply all the possible options one by one and decide which one is the most appropriate. As an applied aspect (i.e., pattern) changes not only the process view but also the goal and performance views, it is possible to observe the impact of the applied pattern on all views. The aspectual model of a redesign pattern may even introduce up-to-now unidentified goals to the process model, which the redesign pattern helps to achieve but have not yet been considered by the modeler, contributing further to a more comprehensive process model.

The advantage of modeling redesign patterns with aspects is that the whole pattern, i.e., the characteristics of the pattern but also its impact on the process, goal, and performance views, can be modeled in one properly encapsulated unit. This facilitates the reuse of the pattern in different applications and enables reasoning about the use of the pattern and its composition with other patterns.

A new concept introduced in this paper is that pointcut expressions span goal and process models. Until now, all AoURN pointcut expressions [14, 25, 26, 27] are either just in the process view or just in the goal view of a URN model. We, however, need to cover both goal and process models a) because redesign patterns require both views to be matched to properly describe the identifying characteristics of the pattern and b) to ensure that all views remain aligned with each other. This provides us with the ability to apply the required changes to goal and performance views after applying the redesign pattern to the process view. Such changes in the goal or performance views are required when the redesign pattern eliminates, adds, or updates tasks in the process view. Therefore, the same tasks and the corresponding KPIs should be added, eliminated, or updated in the goal and performance views, respectively.

4 Case Study

The case study is based on the real Data Warehouse Approval Process of a health care provider that assesses requests for access to the health information in the data warehouse based on patient privacy concerns, ethical concerns, as well as technical feasibility and impact.

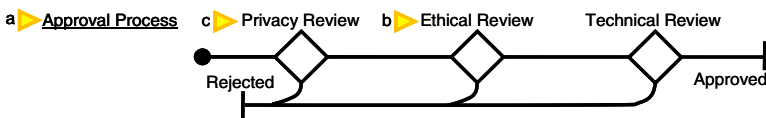


Fig. 6. Improved Data Warehouse Approval Process – Process View

The Knockout redesign pattern reorders a sequence of tasks based on their failure rate and effort. It therefore can be applied to the Data Warehouse Approval Process (see Fig. 2) if the average number of rejections caused by the Privacy Review is higher than for the Ethical Review and the Ethical Reviews takes longer to complete than the Privacy Review. In that case, the Privacy Review should be moved ahead of the Ethical Review to the first task in the sequence (see Fig. 6).

The aspectual model for the Knockout redesign pattern captures in a generic way the constraints of the Knockout pattern as described in the previous paragraph. First, the GRL pointcut expression (GRL graph at the right hand side of Fig. 7) stipulates that there is a value of a KPI in the time category that is not satisfactory (i.e., the top KPI in the GRL pointcut expression as indicated by the T in the circle and <satisfaction, respectively). *Satisfaction* is a value defined by the modeler, usually somewhere between 0 and 100 on the GRL scale. As 0 represents the threshold value and 100 represents the target value, this captures the main premise of the Knockout redesign pattern. As it positively impacts the time category, it should be applied in a situation where a KPI from the time group is not performing as desired.

Furthermore, the unsatisfactory KPI is connected to a task (the one with URN link 3) which is further connected with two other tasks (the ones with URN links 4 and 5). All three tasks are traced with the help of the URN links to the two stubs and the map in the UCM pointcut expression (the map at the bottom left in Fig. 7). Therefore, additional properties from the UCM model must be satisfied for a successful match of the pointcut expression and for the Knockout pattern to be applied. The URN links indicate that the top level task is described by a map and that the other two tasks appear as stubs on the map (i.e., the two other tasks are a refinement of the higher-level task). More specifically, the UCM pointcut expression indicates that the redesign pattern applies to a series of two stubs (i.e., two process steps) that either succeed or fail. Note that the dashed portion of the pointcut expression matches against any sequence of UCM modeling elements and therefore can be matched against the join after the Privacy Review stub in Fig. 2.

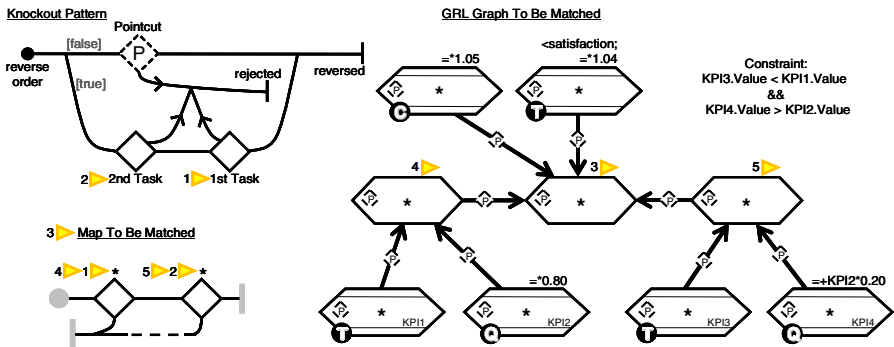


Fig. 7. Generic Aspectual Model for the Knockout Redesign Pattern

At this point, the pointcut expression would match against a large number of consecutive stubs with two out-paths as long as the overall task associated to the map of the two stubs has a time KPI with an unsatisfactory value. To further improve the accuracy of the Knockout pattern description, further matching criteria are defined for the two other tasks. Note that the task with URN link 4 is the first task and the task with URN link 5 is the second task in the sequence as defined by the UCM pointcut expression and the URN links. Each of the two tasks has two more KPIs connected, one from the time category and one from the quality category. Constraints for these four KPIs state that:

- Constraint A: the value of KPI1 (the time KPI of the first task) must be higher than the value of KPI3 (the time KPI of the second task);
- Constraint B: the value of KPI2 (the quality KPI of the first task) must be lower than the value of KPI4 (the quality KPI of the second task).

This reflects the characteristics of the Knockout redesign pattern as it applies only to situations where the second task takes less time to do (time KPI is lower) but results in rejections more often (quality KPI is higher) than the first task. If this is the case, then it is advantageous to move the second task ahead of the first task. This change to the process is described by the Knockout Pattern map (the UCM at the top left in Fig. 7). This map describes the aspectual behavior to be applied if the pointcut expression can be matched for the existing process. AoURN describes a replacement of the matched model elements with an OR-fork with [false] and [true] branches. The [false] branch describes what is being replaced (i.e., the matched model elements represented by the pointcut stub), while the [true] branch describes the new behavior. On the [true] branch the order of the two stubs from the pointcut expression are switched. URN links between the Knockout Pattern map and the pointcut expression allow matched elements to be reused in the Knockout Pattern map (i.e., the aspectual scenario).

Finally, the GRL pointcut expression in Fig. 7 also defines the anticipated impact of applying the Knockout pattern on the performance model by describing the changes to the satisfaction values for matched KPIs. The annotation $=*1.04$ in Fig. 7 indicates that the unsatisfactory KPI is expected to improve by 4%. These changes may be even based on historical data. For example, assuming that 20% of the submitted requests that fail ethical review also fail the privacy review, then the matched quality KPI of the ethical review (KPI2) will decrease by 20% ($=*0.8$) while the matched quality KPI of the privacy review (KPI4) will increase by the same number ($=+KPI2*0.2$), if the order of the two reviews is reversed. Furthermore, there is also an impact on the average cost of the approval process, as more requests will now be rejected earlier in the process leading to a cost decrease.

When the aspect is applied to the URN model of the Data Warehouse Approval Process, a match is found in the performance model. Approval Process, Ethical Review, and Privacy Review match against the three tasks. Avg. Approval Turnaround time, Avg. Ethical Turnaround time, Avg. Ethical Rejections, Avg. Privacy Turnaround time, Avg. Privacy Rejections, and Avg. Cost per Application match against the six KPIs. Furthermore, the map linked to Approval Process and the two stubs linked to Ethical Review and Privacy Review match the UCM pointcut expression.

The pointcut expression for the Knockout redesign pattern is very generic and therefore uses only parameterized elements. This may lead to undesired matches. If this is the case, then the pointcut expressions can be tailored to the specific needs of the current situation. For example, the * for the task with the URN link 3 could be replaced with Approval Process to narrow down the search space.

Since the pointcut expression is matched in the Data Warehouse Approval Process, the aspectual behavior is added to the process (see Fig. 8). Therefore, aspect markers are added before and after the matched model elements as defined on the Knockout Pattern map. AoURN uses slightly different aspect markers to indicate that the aspect replaces existing model elements. Solid bars are added to the aspect markers to denote tunnel entrances and exits. The aspect marker before the Ethical Review stub is a tunnel entrance as the behavior does not continue with the Ethical Review but with the aspectual behavior and only returns to the map at the tunnel exits (i.e., the two other aspect markers). Note how the two stubs on the Knockout Pattern map have been replaced by the matched model elements from the pointcut expression. The resulting model in Fig. 8 is semantically equivalent to the model in Fig. 6.

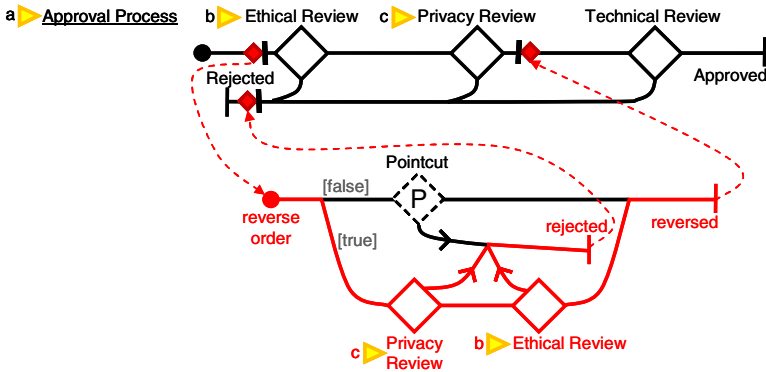


Fig. 8. Applied Knockout Pattern

Two further patterns have been applied to the sub-processes for submitting an application by the researcher and the privacy review by the privacy review board. Fig. 9 shows the “As Is” process and the “To Be” process after the application of the task elimination and control relocation patterns. The first pattern is often used to remove process steps without significant value. In this case, we have removed Review the Results since the average number of mistakes found by this review is low (see Table 2). The second pattern, on the other hand, is often used to move input validation checks to the client side [5]. In this process, we have moved Go over Check List to the researcher’s application submission process. Due to space limitations, we cannot illustrate the aspect models for these two redesign patterns in this paper.

Table 2 shows the impact of all three patterns on the process performance and business goals as calculated by the GRL evaluation mechanism. Although the applied patterns have a positive impact on the Approval Process and Privacy Review performance, they have also a negative impact on the business goals Increase Regulatory

Compliance and Increase Quality of Process. In most cases when redesign patterns are applied, positive and negative impacts occur in parallel [5]. Our suggested approach equips business analysts with the ability to observe and explore these impacts, leading to more informed decisions about the process.

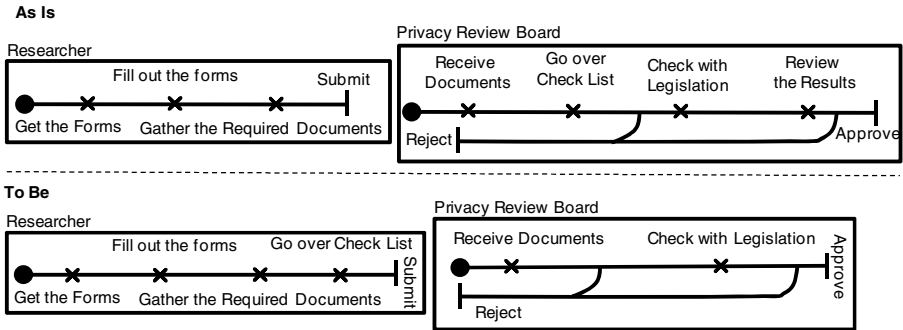


Fig. 9. Applying Task Elimination and Control Relocation to Two Sub-Processes

Table 2. KPI, Goal, and Task Evaluations Before and After Applying the Three Patterns

KPI	SV B	SV A	EV B	EV A	Pattern
Avg. Approval Turnaround Time	7	9	20 d	18.3 d	K – E – CR
Avg. Cost per Application	10	20	4100\$	3800\$	K – E – CR
Avg. Ethical Rejection	90	72	10%	8%	K
Avg. Ethical Turnaround Time	50	50	10 d	10 d	N/A
Avg. Privacy Rejection	80	98	20%	22%	K
Avg. Number of Privacy Mistakes	75	25	0.5%	1.5%	E
Avg. Privacy Turnaround Time	33	66	5 d	4 d	E - CR
Avg. Cost per Privacy Review	50	90	1000\$	800\$	E - CR
Avg. Number of Mistakes	100	N/A	1%	N/A	E
Avg. Cost per Results Review	-50	N/A	200\$	N/A	E
Avg. Rejections due to Missing Docs	-50	75	10%	2%	CR

Goal/Task	SV B	SV A	Pattern	Impact
Increase Quality of Health Care	-3	2	K – E – CR	Positive
Encourage Use of DW	-4	3	K – E – CR	Positive
Increase Approval Response Time	-6	5	K – E – CR	Positive
Reduce Cost	-5	11	K – E – CR	Positive
Reduce Approval Cost	-7	15	K – E – CR	Positive
Increase Regulatory Compliance	74	69	E	Negative
Increase Quality of Process	99	93	E	Negative
Approval Process	30	99	K – E – CR	Positive
Ethical Review	12	12	N/A	N/A
Privacy Review	46	99	E – CR	Positive

SV: Satisfaction value – EV: Evaluation value – B: Before applying redesign pattern – A: After applying redesign patterns

K: Knockout – E: Elimination – CR: Control Relocation

Note: Pattern column indicates which pattern has caused changes in the KPI.

5 Conclusion and Future Work

Although applying redesign patterns to business processes is a natural way to achieve process improvements, existing Business Process Management and Monitoring systems do not provide the means to do so. Business processes, however, have to be improved based on monitoring results to adapt to changes. Furthermore, most business process modeling notations do not support impact analysis of business goals when process model change, since modeling of goals and the relationship between the goals and processes is not supported in the first place.

We have addressed the latter problem in our previous research [10, 11, 13] and further enhanced our proposed methodology in this paper to address the first problem. This paper proposes a framework for the automated suggestion of business process redesign patterns based on monitoring results. Process redesign patterns are modeled individually as aspects which allows for the patterns to be added to and removed from a business process without requiring significant changes to the process models that are difficult to undo, thus enabling the exploration of what-if scenarios. In order to select the most appropriate patterns, several applicable patterns may be applied and their results compared in terms of their impact on the business process and the business goals with the help of the GRL evaluation mechanism.

In our approach, we have utilized AoURN's pointcut expressions and URN KPI groups to model the defining characteristics of a pattern that may be observed in any or all of the process, performance, and goal views. The defining characteristics must be matched before an aspect can be applied to the process model. When the aspect (i.e., a pattern) is applied to the process model, the process, goal, and performance views are updated simultaneously and therefore remain synchronized. To the best of our knowledge, pointcut expressions that span various types of models as required in our approach are a novel idea for AOM (Aspect-oriented Modeling).

In future work, we intend to further automate the selection of the best pattern by applying all applicable patterns and comparing their impact on the process, goal, and performance views. Furthermore, the framework can likely be used to support adaptive business processes and architectures. Additional experiments on real process modifications will also enable us to better validate this approach and further generalize it to be adapted to other modeling languages.

Acknowledgments. This research was partially supported by NSERC through its programs of Discovery Grants and Postgraduate Scholarships.

References

1. van Herk, D.: Business Activity Monitoring Buzz or Business. Master Thesis Information Science, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands (May 2006)
2. Debevoise, T.: Business Process Management with a Business Rules Approach. In: Business Knowledge Architects, Canada (2005)
3. Pourshahid, A., Chen, P., Amyot, D., Peyton, L., Ghanavati, S., Weiss, M., Forster, A.: Toward an integrated User Requirement Notation framework and tool for Business Process Management. In: International MCETECH Conference on e-Technologies, pp. 3–15. IEEE, Washington (2008)

4. Pedrinaci, C., Domingue, J., Medeiros, A.: A Core Ontology for Business Process Analysis. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) *ESWC 2008*. LNCS, vol. 5021, pp. 49–64. Springer, Heidelberg (2008)
5. Reijers, H.: Process Design and Redesign. In: ter Hofstede, A.H., Dumas, M., van der Aalst, W.M. (eds.) *Process-Aware Information Systems*, pp. 207–234. John Wiley & Sons, Inc., Hoboken (2005)
6. Forster, F.: The Idea behind Business Process Improvement: Toward a Business Process Improvement Pattern Framework, <http://www.bptrends.com/publicationfiles/04-06-ART-PatternFramework-Forster.pdf> (accessed, February 2009)
7. Bruce Silver Associates: *BPMS Watch Ratings for Q2 (2008)*, <http://www.bpminstitute.org/bpmsreport.html> (accessed, January 2009)
8. Kronz, A.: Managing of Process Key Performance Indicators as Part of the ARIS Methodology. In: Scheer, A.-W., Jost, W., Heß, H., Kronz, A. (eds.) *Corporate Performance Management*, pp. 31–44. Springer, Heidelberg (2006)
9. Chen, P.: Goal-oriented business process monitoring: An approach based on user requirement notation combined with business intelligence and web services. M.Sc. thesis, SCS, Carleton University, Ottawa, Canada (2007)
10. Pourshahid, A., Chen, P., Amyot, D., Peyton, L., Ghanavati, S., Weiss, M., Forster, A.: Business Process Management with User Requirement Notation. In: *Electronic Commerce Research*. Springer, Heidelberg (to appear, 2009)
11. Pourshahid, A.: A URN-Based Methodology for Business Process Monitoring. M.Sc. thesis, EBT, University of Ottawa, Ottawa, Canada (2008)
12. ITU-T: Recommendation Z.151 (11/08): User Requirements Notation (URN) - Language definition. ITU-T, Geneva, Switzerland (2008)
13. Pourshahid, A., Chen, P., Amyot, D., Weiss, M., Forster, A.: Business Process Monitoring and Alignment: An Approach Based on the User Requirements Notation and Business Intelligence Tools. In: 10th Workshop on Requirement Engineering (WER 2007), Toronto, Canada, pp. 80–91 (2007)
14. Mussbacher, G.: Aspect-Oriented User Requirements Notation: Aspects in Goal and Scenario Models. In: Giese, H. (ed.) *MODELS 2008*. LNCS, vol. 5002, pp. 305–316. Springer, Heidelberg (2008)
15. Amyot, D.: Introduction to the User Requirements Notation: Learning by Example. *Computer Networks* 42(3), 285–301 (2003)
16. Weiss, M., Amyot, D.: Designing and Evolving Business Models with URN. In: Montreal Conference on E-Technologies (MCETECH), Montreal, pp. 149–162 (2005)
17. Weiss, M., Amyot, D.: Business Process Modeling with URN. *International Journal of E-Business Research* 1(3), 63–90 (2006)
18. Mussbacher, G., Amyot, D.: Assessing the Applicability of Use Case Maps for Business Process and Workflow Description. In: International MCETECH Conference on e-Technologies, pp. 219–222. IEEE CS, Washington (2008)
19. Liu, L., Yu, E.: Designing information systems in social context: a goal and scenario modelling approach. *Information Systems* 29(2), 187–203 (2004)
20. Roy, J.-F., Kealey, J., Amyot, D.: Towards Integrated Tool Support for the User Requirements Notation. In: Gotzhein, R., Reed, R. (eds.) *SAM 2006*. LNCS, vol. 4320, pp. 198–215. Springer, Heidelberg (2006)
21. Kealey, J.: Enhanced Use Case Map Analysis and Transformation Tooling. M.Sc. thesis, SITE, University of Ottawa, Ottawa, Canada (October 2007)

22. Roy, J.-F.: Requirement Engineering with URN: Integrating Goals and Scenarios. M.Sc. thesis, SITE, University of Ottawa, Ottawa, Canada (March 2007)
23. Ghanavati, S.: A compliance Framework for Business ProcessesBased on URN. M.Sc. thesis, SITE, University of Ottawa, Ottawa, Canada (May 2007)
24. Pourshahid, A., Peyton, L., Ghanavati, S., Amyot, D., Chen, P., Weiss, M.: Model-Based Validation of Business Processes. In: Shankararaman, V., Zhao, J.L., Lee, K.K. (eds.) Business Process Management: Concepts, Technology, and Application. Advances in Management Information Systems. M.E. Sharpe Inc. (2009)
25. Mussbacher, G., Amyot, D., Araújo, J., Moreira, A., Weiss, M.: Visualizing Aspect-Oriented Goal Models with AoGRL. In: Second International Workshop on Requirements Engineering Visualization. IEEE, Washington (2007)
26. Mussbacher, G., Amyot, D., Weiss, M.: Visualizing Early Aspects with Use Case Maps. In: Rashid, A., Aksit, M. (eds.) Transactions on AOSD III. LNCS, vol. 4620, pp. 105–143. Springer, Heidelberg (2007)
27. Mussbacher, G., Amyot, D., Whittle, J., Weiss, M.: Flexible and Expressive Composition Rules with Aspect-oriented Use Case Maps (AoUCM). In: Moreira, A., Grundy, J. (eds.) Early Aspects Workshop 2007 and EACSL 2007. LNCS, vol. 4765, pp. 19–38. Springer, Heidelberg (2007)

Integration Testing of Web Applications and Databases Using TTCN-3

Bernard Stepien and Liam Peyton

School of Information Technology and Engineering,
University of Ottawa, Canada
{bernard, lpeyton}@site.uottawa.ca

Abstract. Traditional approaches to integration testing typically use a variety of different test tools (such as HTTPUnit, Junit, DBUnit) and manage data in a variety of formats (HTML, Java, SQL) in order to verify web application state at different points in the architecture of a web application. Managing test campaigns across these different tools and correlating intermediate results in different formats is a difficult problem which we address in this paper. In particular, the major contribution of this paper is to demonstrate that a specification-based approach to integration testing enables one to define integration test campaigns more succinctly and efficiently in a single language/tool and correlate intermediate results in a single data format. We also evaluate the effectiveness of TTCN-3 (a standards-based test specification language and framework) in supporting such an approach.

Keywords: web applications, integration testing, databases, TTCN-3.

1 Introduction

Complex web applications in a service-oriented architecture may have to integrate data from several data sources and may have to maintain state in a distributed fashion across many components of the web application. One of the aims of integration testing is to verify intermediate results at key interaction points within the architecture of the web application. This can be done by testing the web application state as captured either in persistent data stored in a data source, or as session data maintained in memory by the web application or any of its distributed components. This can be a complex and challenging task even under the most ideal circumstances.

Traditional approaches to integration testing would typically use a variety of different test tools (such as HTTPUnit, Junit, DBUnit) and manage data in a variety of formats (HTML, Java, SQL) in order to verify web application state at different points in the architecture of a web application. Managing test campaigns across these different tools and correlating intermediate results in different formats is a difficult problem which we address in this paper. In particular, the major contribution of this paper is to demonstrate that a specification-based approach to integration testing enables one to define integration test campaigns more succinctly and efficiently in a single language/tool and correlate intermediate results in a single data format. We also evaluate

the effectiveness of TTCN-3 (a standards-based test specification language and framework [1]) in supporting such an approach.

Using the TTCN-3 specification language, we define an abstract data layer which can maintain web application state across a variety of test tools and formats and verify intermediate results based on tests which transform that abstract data layer. The approach is implemented in a TTCN-3 test framework which uses a collection of test adaptors to mediate between the abstract test layer in which test specifications are defined and the concrete test layer which interacts directly with the web application and its components. Specific examples based on an Online Book Store and sample TTCN-3 specifications are used to verify and correlate database behavior and web application component behavior as they relate to web application state.

TTCN-3 is a test specification and test implementation language for testing distributed systems developed by the European Telecommunications Standards Institute (ETSI). It provides powerful abstraction mechanisms for interfacing to different data and presentation formats and for defining test cases at different levels of abstraction, much as developers use modeling languages to specify the design of a system at different levels of abstraction. This enables reuse across different levels of test activities [2] and the coordination and synchronization of test activities with development activities throughout the development life cycle.

The need for a systematic test framework reflective of web application architecture rather than a patchwork of tools and test scripts has been pointed out as well in other work [3] outside of the TTCN-3 community. Other approaches to integration testing have focused on ensuring formal conformance to web service protocols in web applications that leverage web services as components [4]. TTCN-3 has also been used in this manner [5].

An alternative approach taken to address the low level of detail at which current tools operate is to do model-based testing where test scripts are generated from models. This was done in the AGEDIS case studies [6] where HTTPUnit and HTMLUnit scripts were generated from UML models. In [7] User Requirements Notation (URN), an ITU standard for requirements modeling in telecommunications was used to test web applications. And in [8] evaluations done with JML-JUnit used JUnit scripts generated from JML models of Java classes. Such approaches do link test script generation to an abstract view of the system being tested, but they do not give the same power and flexibility as a test specification approach to verify application logic and information management independent of volatile implementation and presentation details.

2 Book Store Web Application Example

Figure 1 gives a simple example of a typical J2EE web application that supports an on-line book store. We will use this example, throughout the paper to illustrate our approach. The browser interface contains a rich set of HTML, XML, JavaScript, images, stylesheets, etc, that it receives from the web application in response to HTTP requests. The web application, in return, interacts with a variety of components within a service-oriented architecture. It interacts with the book order database via JDBC to keep track of available books and purchases. It interacts with a shopping cart enterprise java bean

via RMI while the customer is shopping online and it interacts with an order processing service via SOAP to let the warehouse know when there is an order of books to ship.

There is also a test framework (implemented in TTCN-3) which can communicate directly via concrete test adaptors with either the web application or any of the components used by the web application. It performs integration testing that verifies intermediate results in terms of application web state based on abstract test specifications which define an abstract data layer in terms of data types, and uses templates for expected responses.

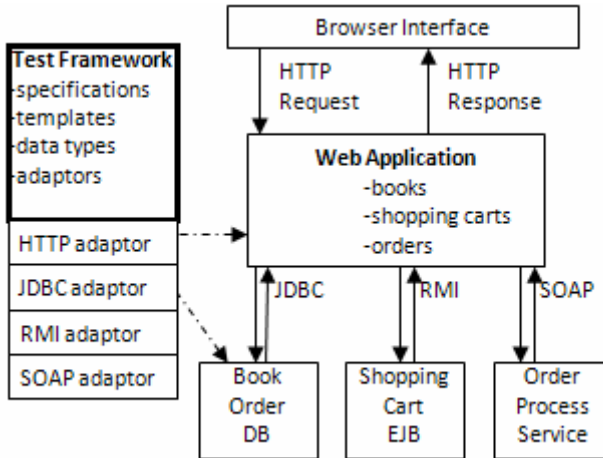


Fig. 1. Book Store and Test Agent

For the purposes of explaining the specification-based approach to integration testing, we will focus on first verifying intermediate results in the Book Order DB, and then we will focus on how to correlate and integrate those results into verification of intermediate results within the web application itself.

3 Database Integration Testing

Specifying test suites for testing database integration mostly consists of specifying test oracles for results of queries or test oracles to verify the state of a database after some update operation. In both cases, testing consists in performing a query and verifying the result set. The central TTCN-3 concept to represent test oracles is the template. In sharp contrast with traditional testing methodologies, a TTCN-3 template performs the checking of all the data involved in a result set in a single operation. The setting of an oracle is organized in three steps:

- The definition of a data type to represent the results set
- The definition of a template of values for each element of the data type to be matched.
- The definition of a test events sequence with possible alternative paths.

In contrast to unit testing approaches, TTCN-3 is more behavior oriented. This includes being able to correlate the results of several consecutive events where the current state of the database is dependent on all previous steps.

3.1 Data Typing

Abstract data types to represent database results are defined in terms of results sets as sets of rows where each row is an individual database record and columns correspond to fields. Union in TTCN-3 can be used to define results that join elements from different tables. For example, below we define a BooksTable, PublisherTable and the join between them to create a CatalogEntry result set.

```

type record DBBooksTableType {
    charstring author,
    charstring title,
    float price
}

type record DBPublisherTableType {
    charstring pubName,
    charstring street,
    charstring city
}

type record BooksPublisherResultType {
    charstring pubName,
    charstring author,
    charstring title
}

```

The three above basic types are merged into a TTCN-3 union type as follows:

```

type union CatalogEntry {
    DBBooksTableType booksTable,
    DBPublisherTableType publisherTable,
    BooksPublisherResultType bookspubResult
}

```

Finally, rows are represented using the TTCN-3 set of type construct on the previously defined union.

```

type set of CatalogEntry CatalogEntryResultSet;

```

3.2 Specifying Test Oracles Using TTCN-3 Templates

The TTCN-3 template is more than an instance of test data for a given data type. It looks like a plain assignment of values but is in fact capable of complex matching. Values can instead be list of alternate values, ranges for numeric values or pattern matching specifications for strings. Thus, a TTCN-3 template is more like a hybrid between a data assignment and some potentially complex logical expression. Since it can be parametric it actually serves the role of a function. The most important concept however remains that the actual matching mechanism is built-in and thus needs zero coding effort from the tester. An atomic element of a test oracle can be represented by

the following template where the field `booksTable` indicates the type among the types of the union used and the assigned values indicate the oracle's criteria.

```
template CatalogEntry amerique := {
    booksTable := {
        author := "Herge",
        title := "Tintin en Amerique",
        price := 8.20
    }
}
```

The template is also a powerful structuring concept since it can be re-used by other templates. For example, a list of database items defined individually as above can be re-used to define a list of items, thus different database rows, as follows:

```
template CatalogEntryResultSet myBooks := { templeSoleil, ileNoire,
amerique };
```

Zoio recommends to externalize assertions [9] by avoiding the scattering of hard coded assertions throughout the test code and also to write comprehensive tests that cover all aspects of a database state and finally use precise assertions. The TTCN-3 template concept naturally implements all of these recommendations.

3.3 Performing a Test

In TTCN-3, a database test is specified by sending an SQL request over a communication channel that is abstracted as a TTCN-3 port and by receiving a response over that same channel. The receive statement actually serves two purposes: to obtain data from the communication channel and to match this obtained data with a template. In the following example, after sending an SQL request, we attempt to match the result set to the template `myBooks` that we defined previously.

```
db_port.send("select * from books");
alt {
    [] db_port.receive(myBooks) {setverdict(pass)}
    [] db_port.receive {setverdict(fail)}
}
```

The above code illustrates how test verdicts are set by using the TTCN-3 `alt` construct. The first case corresponds to the expected response being received and the test verdict being set to pass. The second alternative case consists in receiving anything else which would result in setting the test verdict to fail instead. The TTCN-3 `alt` construct is a powerful concept that enables one to specify complex test behaviors through nesting as trees that represent various possible sequences of test events with their corresponding test verdicts in the leafs of the tree.

3.4 Separation of Concerns Between Abstract and Concrete Layer

So far we have only defined a high level abstract test specification without specifying how the data is actually obtained. This is intentional because one of the central premises of TTCN-3 is that there should be an explicit separation of concerns between the

abstract and concrete layers. The abstract test specification is defined solely at the abstract layer. The concrete layer (also called the test adapter layer) is where the actual connection with a database occurs and where the data is retrieved from results set specified in a general purpose language (GPL). However, the actual classes and member functions for the test adapter are fully defined in the TTCN-3 standard, part 5 [10]. There is a correspondence between the abstract layer `send` command and the test adapter's `triSend` method. This is where the typical JDBC [11] database interface would take place and, at this point, nothing is unusual compared to the traditional GPL test implementation, as can be observed in the following example of the implementation of the `triSend` method.

```
public TriStatus triSend(final TriComponentId componentId,
    final TriPortId tsiPortId, final TriAddress address,
    final TriMessage sendMessage) {

    byte [] mesg = sendMessage.getEncodedMessage();
    if(tsiPortId.getPortName().equals("system_dbPort")) {
        String theSQLRequest = new String(mesg);
        Connection db_connection = null;

        try {
            Class.forName ("com.mysql.jdbc.Driver").newInstance ();
        } catch ...

        try {
            String url = "jdbc:mysql://localhost/ebookstore";
            db_connection = DriverManager.getConnection (url, null, null);
        } catch (SQLException e) { ... }

        try {
            Statement db_statement = db_connection.createStatement();
            boolean status = db_statement.execute(theSQLRequest);
        } catch ...

        ResultSet theCurrentResultsSet = db_statement.getResultSet();
    } }
```

The results of the statement's execution would then be retrieved and transformed into abstract data by a codec (coder/decoder). The codec is part of the concrete (or test adapter layer). For that purpose, the result set object instance is serialized so that it can be passed as a `byte[]` stream to the codec.

```
byte[] theByteRepresentation = ((DBCodec)
    getCodec("")).serializeObject(theCurrentResultsSet);
```

The codec can be written in different ways. Normally, there is a corresponding codec for each abstract data type. For JDBC, we have found a more generic approach for a codec that can handle any abstract data type without having to know what data types are used in a test suite. Thus, this codec is a perfect framework that can be used in any database testing application. Its full description can be found in [12]. The separation of concerns of TTCN-3 has some additional benefits of re-usability of the abstract layer across platforms and implementation languages.

4 Integration Testing of Web and Database Applications

So far, we have shown examples of test specifications that, despite their abstractedness, are not too different from unit testing since they involve only the database. The need to test databases in conjunction with the web application that uses them has been pointed out in [13]. They report on a tool called AGENDA that produces test paths using a cyclomatic complexity algorithm. It is based on a white box approach and addresses three concerns: better coverage, more appropriate input values for forms and better targeting of test efforts. However, they use plain XML files to assemble their test specification which unfortunately adds some unnecessary complexity to the problem.

The real value of using TTCN-3 is beyond mimicking unit testing and instead is found in the specification of complex systems that consist of various components that perform different services, some being database services and others being web services or user interface services such as presenting web pages. Combining such composite services into a single integration test can be challenging when using a GPL. This is mostly due to the frequent tendency to mix test assertions and data extraction functionalities. The separation of concerns that TTCN-3 supports enables us to specify test suites strictly at the abstract level and thus enable the tester to focus on the purpose of the test.

A frequent class of applications consists in the combination of web applications with databases. Here, two kinds of tests can be performed:

- Check the database state after a web user submitted data through a web application.
- Check the results after a user did a query to the database over a web application to see if they correspond to the state of a database.

4.1 Consistency Check Between Web Data Entries and Database State

Web data entry is achieved by submitting web forms that have been filled with data. Thus, in order to specify the test to check the consistency between the web data entries and the resulting database state, we need to handle both aspects of the integration test and first how to submit a form in an abstract way and eventually how to translate this abstract request into a concrete web query. A complete description of various approaches to achieve the above has been presented elsewhere [14]. Here we will briefly show the essential abstract layer elements required to specify an HTML form so as to be able to illustrate the concept of test oracle transformation later.

```

type record ParameterValueType {
    charstring parmName,
    charstring parmValue
}

type set of ParameterValueType ParameterValuesSetType;

type record FormSubmitType {
    charstring formName,
    charstring buttonName,
    charstring actionValue,
    ParameterValuesSetType parameterValues
}

```

Using the above abstract data type, we can specify TTCN-3 templates for web form submissions. First a definition of a filled web form for entering a specific book:

```
template ParameterValuesSetType filledFormAmerique := {
  {parmName := "author", parmValue := "Herge"},
  {parmName := "title", parmValue := "Tintin en Amerique"},
  {parmName := "price", parmValue := "8.00"}
}
```

Then we specify a parametric template to describe the form itself using a formal parameter to indicate the actual form parameters values for a specific book. This template can be re-used to submit an arbitrary number of different books.

```
template FormSubmitType webInsertionFormSubmit
  (ParameterValuesSetType theParameters) := {
  formName := "bookAdditionForm",
  buttonName := "add",
  actionValue :=
    "http://localhost:8080/eBookStore/servlet/book_insertion",
  parameterValues := theParameters
}
```

Finally, we specify the typical test behavior statement that executes this form submission, namely a send command with the parametric template fully instantiated with the previously defined template about the elements of the book being inserted in the database and finally a receive statement that attempts matching the web response to yet another template defining the expected web response.

```
web_port.send(webInsertionFormSubmit(filledFormAmerique));
web_port.receive(webResponsePage);
```

The test adapter layer's codec then produces the appropriate web request as follows:

```
http://localhost:8080/eBookStore/servlet/book_insertion?author=Herge&title=Tintin%20en%20Amerique&price=8.00
```

This web request would then be submitted on a TCP/IP channel using a post command. This example also illustrates how the TTCN-3 template achieves another separation of concern between test behavior and conditions governing behavior.

At this point we have submitted the filled form and all we need to do is to perform a test on the database to see if the data has been stored using the test described in section 2. However, this would be a kind of double hard coded test oracle approach. We certainly cannot avoid hard-coding the form submission since we need some starting point; we could, however, avoid hard-coding the test oracle for the database results by merely transforming the form submission template into a database result set template to check the state of the database. This is possible in TTCN-3 because of its ability to specify dynamic templates that are constructed from other tests results. The most important fact is that TTCN-3 can do such a transformation without having the

data encoding or extraction required in a GPL. Thus, this transformation can be achieved relatively concisely at the abstract level as in the following example:

```
function transformForms2DB(FormsParametersValuesSetType theFormParms)
return CatalogEntryResultSet {
  ...
  for(i:=0; i < numOfForms; i:=i+1) {
    anItem.booksTable.author :=
      getFieldValue("author", theFormParms[i]);
    anItem.booksTable.title :=
      getFieldValue("title", theFormParms[i]);
    anItem.booksTable.price :=
      str2float(getFieldValue("price", theFormParms[i]));
    theItems[i] := anItem;
  }
  return theItems;
}
```

Thus, a complete integration test can now be specified as follows:

```
testcase web2DatabaseResultsTest() runs on MTCType system SystemType {
  var DBSelectResponseType theDBSelectResponse;

  map(mtc:dbPort, system:system_dbPort);
  map(mtc:webPort, system:system_webPort);

  // database re-initialization
  dbPort.send("delete from books");

  // have a user insert a book through a web page form
  webPort.send(webInsertionFormSubmit(filledFormAmerique));
  ...
  // transform the list of filled forms information into
  // a database query results template
  var CatalogEntryResultSet expectedDatabaseResults :=
    transformForms2DB({filledFormOrNoir, filledFormAmerique});

  // check if the database contains the entered books
  dbPort.send(myBooksSelectRequest);
  alt {
    [] dbPort.receive(myBooksSelectResponse(expectedDatabaseResults))
    {
      setverdict(pass)
    }
    [] dbPort.receive {
      setverdict(inconc);
    }
  } } }
```

4.2 Consistency Check Between Database State and Web Queries

Given a specific state of the database, we define a test that consists in simulating a user performing a query through a web page and obtaining data that is displayed on the response page. The second step of the test consists in performing a direct SQL database query to obtain the same data as through the web page and compare it to the data obtained through the web page. If the two sources of data coincide, the test has passed.

The second step of this test is identical to the second step of the previous test (web data insertion against database query). We can re-use the same SQL statement for that purpose. These SQL statements can be extracted from the application under test as suggested in [15]. They propose a testing approach that transforms the embedded SQL statements in database applications to procedures in a general-purpose programming language (GPL). Here we replace the GPL with TTCN-3 and gain clarity and conciseness. The first step however is somewhat similar since we need to submit a form with some pre-filled fields, this time with the parameters of the query and with the different requested actions as follows:

```
template FormSubmitType queryBooksHerge := {
  formName := "queryForm",
  buttonName := "query",
  actionValue :=
    "http://localhost:8080/eBookStore/servlet/book_selection",
  parameterValues := {
    {parmName := "author", parmValue := "Herge"},
    {parmName := "maxPrice", parmValue := "10.0"}
  }
}
```

Again, this web query is submitted to the web application using a TTCN-3 send command as follows:

```
webPort.send(queryBooksHerge);
```

This web query will result in a web response page that we need to specify using TTCN-3 abstract data types and templates. A full description on how to achieve this can be found elsewhere [14]. Here we summarize some main ideas. A web page is modeled using the following types:

```
type record WebPageType {
  integer statusCode,
  charstring title,
  charstring content,
  LinkListType links optional,
  FormSetType forms optional,
  TableSetType tables optional
}
```

Web page data is typically displayed using HTML tables that can be modeled with the following TTCN-3 types:

```
type set of charstring RowCellSetType;

type record TableRowType {
  RowCellSetType cells
}

type set of TableRowType TableRowSetType;

type record TableType {
  TableRowSetType rows
}

type set of TableType TableSetType;
```

Once the types are defined, we can define the parametric template for the web response that is composed of constants such as the page title and the status and a parameter for the actual tables containing the requested data.

```
template WebPageType
  hergeDBQueryResultsPage(TableSetType theTables) := {
    statusCode := 200,
    title := "bookstore.com query items page results",
    content := ?,
    links := {},
    forms := {},
    tables := theTables
  }
```

Here again, we could have hard coded the values of the tables but, instead, in order to avoid duplicate work we prefer to dynamically create it by deriving it from the result set of the database query using a function as follows:

```
function transformDBResultsIntoHTMLTables(ItemsType theDBItems)
  return TableSetType {
  ...
  theTableRows[0] := { cells := {"author", "title", "price" } };

  for(i:=0; i < numOfDBRows; i:=i+1) {
    if(ischosen(theDBItems[i].booksTable)) {
      aBook := theDBItems[i].booksTable;
      aRow := {
        cells := { aBook.author, aBook.title,
                  myFloat2str(aBook.price) }
      };
      theTableRows[i+1] := aRow;
    }
  }

  theTable := { rows := theTableRows };
  tables[0] := theTable;
  return tables
}
```

Finally the full test behavior is specified as follows:

```
testcase database2webResultsTest() runs on MTCType system SystemType {
  var DBSelectResponseType theDBSelectResponse;

  map(mtc:dbPort, system:system_dbPort);
  map(mtc:webPort, system:system_webPort);

  ... // set the database in the desired state

  dbPort.send(myBooksSelectRequest);
  dbPort.receive(myBooksSelectResponse(myBooks))
    -> value theDBSelectResponse {
  var CatalogEntryResultSet theReceiveDBItems :=
    theDBSelectResponse.items;

  var TableSetType booksTables :=
    transformDBResultsIntoHTMLTables(theReceiveDBItems);

  webPort.send(queryBooksHerge);
```



```
alt {
  [] webPort.receive(hergeDBQueryResultsPage(booksTables)) {
    setverdict(pass)
  }
  [] webPort.receive {
    setverdict(fail)
  }
}
```

5 Conclusions and Future Work

In this paper, we have demonstrated two main advantages of a test specification approach for integration testing. First, test cases can be defined much more succinctly using a single common language. This simplifies not only the writing of test cases, but also the reading and understanding of these test cases. It also eliminates the need to consult and understand test cases defined and written in several different languages. Secondly, and perhaps more importantly it enables intermediate results that are communicated using different data formats and protocols, to be integrated, combined, compared and verified within a single, consistent data abstract layer.

We have also demonstrated the suitability of TTCN-3 both as a test specification language and as a framework for executing integration tests. It supports the definition of an abstract specification layer separate from test adaptors which manage implementation specifics. TTCN-3 templates that are used to specify test oracles are created dynamically based on defined abstract transformations between web requests and the virtual data layer. The virtual data layer is mapped to different database tables or views by a universal data codec.

While we have focused on integration testing in this paper, the approach can also be used for blackbox and white box testing related to databases and session state. Black box testing using parallel testing is proposed in [16]. In particular, they recommend to avoid the traditional approach of resetting the state of a database before each test as is often recommended [9] because this is a time consuming process and also because it does not reflect the realities of a multi-user application in general. In TTCN-3, we have already shown the benefits of multi-user application testing in [17] and believe the extension of these principles to databases should be straightforward.

Whitebox testing as described in [18] can also be implemented in a straight forward fashion at an abstract level using TTCN-3. They state that the full behavior of a database application program is described in terms of the manipulation of two very different kinds of state: the program state and the database state. While, so far, we have used a message oriented approach in our abstract test suites, TTCN-3 provides also a procedure oriented approach. It can be used to invoke functions or methods of the application under test directly and, thus, check the resulting state of both the software and the database.

Acknowledgements

The authors would like to thank Testing Technologies IST GmbH for providing us the necessary tool -- TWorkbench -- to carry out this research as well as NSERC for partially funding this work.

References

1. ETSI ES 201 873-1, The Testing and Test Control Notation version 3, Part1: TTCN-3 Core notation, V3.4.1 (September 2008)
2. Probert, R.L., Xiong, P., Stepien, B.: Life-cycle E-Commerce Testing with OO-TTCN-3. In: FORTE 2004 Workshops proceedings (September 2004)
3. Rankin, C.: The Software Testing Automation framework. *IBM Systems Journal, Software Testing and Verification* 41(1) (2002)
4. Bertolino, A., Frantzen, L., Polini, A., Tretmans, J.: Audition of web services for testing conformance to open specified protocols. In: Reussner, R., Stafford, J.A., Szyperski, C. (eds.) *Architecting Systems with Trustworthy Components*. LNCS, vol. 3938, pp. 1–25. Springer, Heidelberg (2006)
5. Stepien, B., Schieferdecker, I.: Automated Testing of XML/SOAP based Web Services. In: Proc. of the 13th. Fachkonferenz der Gesellschaft für Informatik (GI) Fachgruppe KiVS (February 2003)
6. Craggs, I., Sardis, M., Heuillard, T.: AGEDIS Case Studies: Model-based Testing in Industry. In: Proc. 1st European Conf. on Model Driven Softw. Eng., Nuremberg, Germany, imbus AG, December 2003, pp. 106–117 (2003)
7. Amyot, D., Roy, J.-F., Weiss, M.: UCM-Driven Testing of Web Applications. In: *SDL Forum* (2005)
8. Tan, R.P., Edwards, S.H.: Experiences Evaluating the Effectiveness of JML-JUnit Testing. *ACM SIGSOFT Software Engineering Notes* 29(5) (September 2004)
9. Zoio, P.: Testing 1,2,3.... Oracle Magazine (July-August, 2005),
<http://www.oracle.com/technology/oramag/oracle/05-jul/o45testing.html>
10. ETSI ES 201 873-5 V3.3.1, The Testing and Test Control Notation version 3; Part 5: TTCN-3 Runtime Interface (TRI) (April 2008)
11. JDBC, <http://java.sun.com/docs/books/tutorial/jdbc/index.html>
12. Stepien, B.: A generic TTCN-3 codec framework for testing Database applications, Working Paper, School of Information Technology and Engineering, University of Ottawa (2008)
13. Deng, Y., Frankl, P., Wang, J.: Testing Web Database Applications. *ACM SIGSOFT Software Engineering Notes* 29(5), 1–10 (2004)
14. Stepien, B., Peyton, L., Xiong, P.: Framework Testing of Web Applications using TTCN-3. *International Journal on Software Tools for Technology Transfer* 10(4), 371–381 (2008)
15. Chan, M.Y., Cheung, S.C.: Testing Database Applications with SQL Semantics. In: *Proceedings of 2nd International Symposium on Cooperative Database Systems for Advanced Applications, CODAS 1999* (1999)
16. Binnig, C., Kossmann, D., Lo, E.: Testing Database Applications. In: *Proceedings of the 2006 ACM SIGMOD international conference on Management of data* (2006)
17. Peyton, L., Stepien, B., Seguin, P.: Integration Testing of Composite Applications. In: *Proceedings of the 41st Hawaii International Conference on System Sciences, HICSS 2008* (2008) ISSN:1530-1605,
<http://csdl.computer.org/comp/proceedings/hicss/2008/3075/00/30750096.pdf>
18. Willmor, D., Embury, S.M.: Exploring test adequacy for database systems. In: *Proceedings of the 3rd UK Software Testing Research Workshop* (September 2005)

A Reference Model for Semantic Peer-to-Peer Networks

Abdul-Rahman Mawlood-Yunis¹, Michael Weiss², and Nicola Santoro¹

¹ School of Computer Science, Carleton University,
{armyunis,santoro}@scs.carleton.ca

² Department of Systems and Computer Engineering, Carleton University
weiss@sce.carleton.ca

1125 Colonel By Drive, Ottawa, ON, Canada K1S 5B6

Abstract. Today's information systems are highly networked and need to operate in a global world. With this comes the problem of semantic heterogeneity of information representations. Semantic peer-to-peer networks have been proposed as a solution to this problem. They are based around two components: a peer-to-peer infrastructure for information exchange between information system, and the use of ontologies to define application semantics. However, progress in this area is hampered by a lack of commonality between these approaches, which makes their comparison and translation into practical implementations difficult. In this paper, we describe a reference model for semantic peer-to-peer networks in an effort to remedy this problem. The reference model will (1) enable the establishment of a common terminology for describing semantic peer-to-peer networks, and (2) pave the way for an emerging standardized API that will promote information system interoperability.

Keywords: System modeling, Interoperability, P2P, Ontology, Information system.

1 Introduction

Today's information systems are highly networked and need to operate in a global world. With this comes the problem of semantic heterogeneity of information representations. Semantic peer-to-peer networks (SP2P) have been proposed as a solution to this problem. They are based around two components: a peer-to-peer infrastructure for information exchange between information system, and the use of ontologies to define application semantics.

SP2P systems have several subtypes as shown in Table 1. They include P2P knowledge management systems, P2P databases, P2P Semantic Web, P2P emergent semantics systems, P2P information systems, and P2P Web Services. However, progress in this area is hampered by a lack of commonality between these approaches, which makes their comparison and translation into practical implementations difficult. The lack of commonality is mainly due to the different backgrounds of researchers (knowledge management, databases, information retrieval, P2P, etc.) and the still nascent state of the field.

Table 1. SP2P System Types

SP2P Types	System Instances
P2P knowledge management systems	KEx [6]
P2P databases	coDB [16] , Piazza [24] , PeerDB [34] , Hyperion [26]
P2P Semantic Web	BiBster [21] , Somewhere [37] , P2PSW [39]
P2P emergent semantics systems	Chatty Web [3] , DisES [15]
P2P information systems	P2PSLN [23] , Observer [32] , P2PISM [40]
P2P Web Services	ESTEEM [5]

In this work, we describe a reference model for SP2P networks in an effort to model the emerging decentralized computing paradigm in a generic and high level abstraction. The potential contribution of the reference model to the advancement of the current SP2P networks spans over various areas. These include: 1) an establishment of common terminologies for describing SP2P networks. This leads to a better understanding and communication among members of the community. 2) empowering users to assess the quality of existing SP2P systems. System qualities could be determined through checking whether or not an individual system implements the features and functionalities that are affirmed by the generic model. 3) enabling quality comparison among individual systems. Individual system could be compared with each other on whether they comply with the generic model, and how they implement the generic affirmed features. 4) paving the way for an emerging standardized API that will promote information system interoperability. In this work, the emphasis is given to the first and last tasks.

The rest of this paper is organized as follows: In Section [2](#), related work will be reviewed, In Section [3](#), some features of SP2P systems are briefly described. In Section [4](#), the key constructs of SP2P networks are discussed along with the interfaces specification for the model, and we draw conclusions in Section [5](#).

2 Related Work

To the best of our knowledge, there are only few works which directly address the problem of building reference models for P2P networks. Some of these works are described here. In [\[1\]](#) a reference model for unstructured P2P networks have been presented. In addition to identifying core components of P2P networks, [\[1\]](#) discusses the network’s essential design decisions. It also provides a brief comparison of some relevant P2P networks. Similarly, a reference model for structured P2P networks have been provided in [\[12\]](#). From high-level abstraction view, we consider this current work to be an extension/adaptation of the mentioned P2P reference models to a new environment. In this new environment semantic aspects play essential roles in molding and building P2P network. Other related

works are [28,38]. In [38] authors show only preliminary steps toward modeling semantic overlay networks. The efforts in [28], on the other hand, is more spend on discussing different query routing strategies rather than generic model. There are also some related works in a closely related domain, i.e. grid domain, for example [35]. These works were helpful for understanding system layers and describing components from high level perspective.

3 Differences Between P2P and SP2P Systems

SP2P systems represent the next step in the evolution of P2P networks because SP2P systems incorporate several additional features not present in P2P networks. As elaborated in detail below, we reviewed existing SP2P systems [3,6,15,23,21,32,26,40] and other research on semantic P2P systems [7,8,22,25,27], and came to the conclusion that there are several features that distinguishes P2P systems from SP2P systems. This include: 1) formally-structured information, 2) local mapping, 3) autonomous peer resource management, and 4) semantic based routing.

Data or information in SP2P systems is *structured* and formal. The purpose of formally-structured data is to enrich data semantics and support inferences which in turn improve search performance and the quality of retrieved information.

The *local mapping* is used as a translational capability to forward queries between the peers under the conditions when the peers possess different data schema or knowledge representations.

Autonomous peer resource management concerns with peers control resources and not losing their autonomy. That is, in contrast to conventional P2P networks, resources in SP2P are neither replicated nor assigned to other peers in the network in order to be used by network peers for processing queries. This is because the focus of SP2P systems are mostly applications where replication of resource is not permissible [26,24]. However, in semantically-enhanced P2P file sharing systems this feature can be relaxed.

Query routing in SP2P systems is different than non-semantic P2P systems. This is mainly due to the fact that SP2P systems are unstructured P2P networks. That is, SP2P systems are different than structured P2P networks such as Chord [11] or Pastry [36] and other distributed hash table based systems. In SP2P, semantic based peer selection (discovery) method relates peers with similar domain knowledge, and these relation links are used for query routing process.

We see the above described system features to be prominent characteristics that differentiate SP2P systems from the conventional P2P systems like [17,20,33].

4 SP2P Reference Architecture

There are many and diverse SP2P system realizations and architectures. This is primarily due to the involvement of a variety of researchers from different backgrounds into a still recent and evolving area. The proposed SP2P reference

architecture models the essential aspects of the existing systems. A particular system can be considered an instance of the reference architecture. The model is a high level abstraction which hides implementation detail from the implementers. However, it is defined in a way which makes deriving concrete systems possible. Systems built based on this model should be easy to change and modify. The SP2P reference model is made of seven key constructs. We obtain the key constructs from a comparative analysis of existing SP2P systems. We studied the features of existing SP2P systems and have identified the commonalities among them to create a construct. The reference model constructs are Peers P , Resources R , Query Formulator QF , Semantic Neighborhood SN , Mapping M , Router T , and Query Answerer QA : $SP2P = \{P, R, QF, SN, M, T, QA\}$. In the following we will describe each of these model constructs.

We would like to emphasize that the model provides the minimal common components and components structures shared by different SP2P systems. For example, in our model, query object is merely concepts, while in a system such as Chatty Web [3] they cover several additional parameters. These include Mapping Path, TTL, etc. Yet, we believe that concrete systems can make use of our model simply through component sub-classing and extension.

4.1 Peers

A Peer $P = \{ID, R, O, N\}$ represents an active object or an entity in the network. Each peer has a unique identification ID , a set of resources R that it manages, a profile O , and a set of neighbors N , i.e. references to other peers in the network. The profile describes peer's domain knowledge and used in peer discovery process. The peer's profile could be a description of its schema, a subset of schema key words, or a description of peer's expertise and services. Figure 1 is a class view of Peer construct for the proposed SP2P reference architecture.



Fig. 1. Peer construct

Examples: $P = \{ID, R, O, N\}$ is an essential system construct for Chatty Web [3], KEx [6], P2PSLN [23] and Piazza [24]. These systems, however, are different on the way N is identified. Further, while Chatty Web and P2PSLN store O explicitly, this construct is implicit in KEx and Piazza.

4.2 Resources

The Resources $R = \{DM, I, MD\}$ is one of the fundamental building blocks of any SP2P system. Peer resources comprise data model DM , the actual data I , and meta-data MD . Peers could have their data represented in different DM . Examples of DM include Relational table, XML Schema, RDF data model, OWL. MD is a link or a reference to external resources available on the network. In contrast to conventional P2P networks, R in SP2P are neither replicated nor assigned to other peers in the network in order to be used by network peers for processing queries¹. The choice of DM is important, and systems could be differentiated from each other based on the choice of their DM . This is due to the following two features of the highly structured data:

- 1) **Support for Semantics.** The choice of data model determines data semantic transparency. Semantic transparency in turn enables automatic machine processing of data as well as improving query result precision (recall).
- 2) **Support for Inferences.** The choice of data model determines the extent of the system's ability to answer queries. For example, data models such as RDF and OWL support knowledge *inferences*. Systems with this types of data modeling are able to answer queries where information is not explicitly stored in their repository. This might be difficult for other systems with different data models to do so.

Figure 2 is a class view of the Resource construct for the proposed SP2P reference architecture.

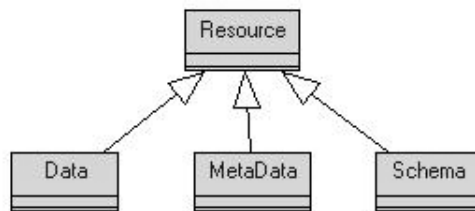


Fig. 2. Resource construct

Examples: The Resources R form a fundamental building block for each of Chatty Web [3], KEx [6], P2PSLN [23] and Piazza [24]. These systems, however, are different from each other on the choice of DM and MD . For example, while P2PSLN's DM is XML, Piazza system supports both XML and RDF. Chatty Web authors, on the other hand, declares that their system is orthogonal to underline data model, and they use XML as the DM for their running example. KEx's resource structure is slightly different than the above mentioned systems. KEx's R comprise collection of documents organized according to local semantic schema and managed by local application.

¹ In cases where SP2P is used for example for file sharing, this feature might not hold.

4.3 Query Formulator

Query Formulator $QF = \{SC, CQ, PQ, q, L\}$, often a graphical user interface component, is a separate component on top of the resource layer. Peers use their own Query Formulator QF to *select concepts* SC from the local resource repositories, *compose queries* CQ and *place queries* PQ on the neighboring peers N . Query objects q are diverse, based on the system's endorsement for the query's explicit semantics, i.e. peer's DM (see subsection 4.2). For example, query content could incorporate references to local or global ontologies for supporting query concept meanings, or when a tree-like data representation is used as a resource, e.g. XML format, a query concept could be replaced by a tree path. A tree path refers to the concept, its ancestors, and descendant concepts. Another important aspect relevant to the query formulation module is the *query language* L . The choice of the L restricts the semantic explicitly of query content. Figure 3 represents a class view of the Query Formulator construct for the proposed SP2P reference architecture.

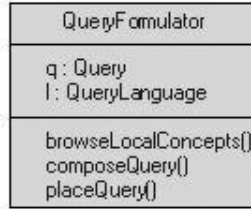


Fig. 3. Query formulator construct

Examples: QF is an independent component in each of Chatty Web, KEx, P2PSLN and Piazza systems. KEx for example, uses QF to perform SC , CQ , and PQ operations, and concept used in q are replaced by XML tree paths in order to explicate their meanings; while in Piazza L is a modified XQuery language, P2PSLN have developed its own L .

4.4 Semantic Neighborhood

Discovering and grouping together peers with compatible semantic information, i.e. forming semantic neighborhood $SN = \{A, V, sim, d\}$, is a distinguishing characteristic of SP2P systems. That is, SP2P network topology is unstructured and semantic based. Two popular methods for forming a semantic neighborhood include:

Autonomous Joining (A). Peers select autonomously which other peers they are going to connect with. Peers are responsible for identifying semantically related peers, and construct semantic mapping(s) between their own information resources (ontology) and ontologies of related peers when their domain representations are different.

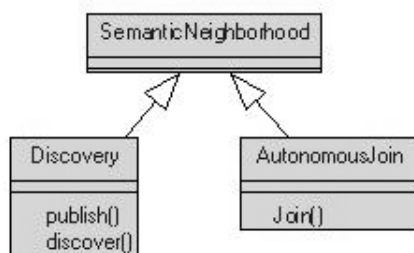


Fig. 4. Semantic neighborhood construct

Peer Discovery (V). Peers exchange their profile O and use similarity function sim to discover semantically related peers. The exchange of O can happen at network startup time or when new peers join an already established semantic based network. Peers interested in connection with other peers, broadcast their O , and relevant peers respond to the querying peer by sending their ID and O . Querying peer computes the **strength** of the relation, i.e. the semantic affinity between the two profiles, and either accepts or drops the answer for connection. Peers could have only limited number of connection. The **network degree** d represents this limitation in the model. Figure 4 represents a class view of the Semantic Neighborhood construct for the proposed SP2P reference architecture.

Examples: SN in Chatty Web system consist of peers with the same schema, and peers with different schemas when peers are able to provide mapping between their schemas. Similarly, in the Piazza, peers' SN comprise of all other peers that are related to their schema by semantic mappings. The Chatty Web and KEx systems are different from P2PSLN and Piazza by the fact that the formers use V to form their SN , but the latter's employ A method.

4.5 Mappings

Semantic Mapping $M = \{ME, MI, MC, MW, MM\}$ refers to semantic relationship between concepts from independent information sources (ontologies) [13]. It is a fundamental design building block for any SP2P System [14], and a topic undergoing heavy research [9]. Using semantic mapping with SP2P systems involve decision making on various issues including mapping expressiveness ME , mapping implementation MI , mapping correctness MC , mapping ownership MW and mapping maintenance MM . Below, a short description of each of these mapping constructs is highlighted.

Mapping expressiveness (ME). Semantic mapping in its simplest form could be just a matter of finding query concept synonyms among different ontologies. In more involved mappings, logical relations are used for finding relationships among concepts, concept properties and attributes.

The set of logical relations commonly used to define relationships among the peers' ontology concepts are $\{\equiv, \sqsubset, \sqsupset, *, \perp\}$. In this case, $c_1 \equiv c_2$ means that the two concepts are synonyms. In other words, c_1 and c_2 are different concepts with similar or identical meanings and are interchangeable. The relation $c_1 \sqsupset c_2$ means c_1 is hypernym of c_2 . That is, c_1 is more generic or broad than c_2 . The relation $c_1 \sqsubset c_2$, means that the c_1 have a hyponym relation to c_2 , i.e. c_2 is more generic or broad than c_1 . The relation \perp means that two concepts have no semantic relation with each other. Any other relations between concepts other than those described above can be captured by $*$ relation.

Mapping expressions have an effect on the extent of query results. They could increase or decrease the extent of query result based on the permissible logical expressions of the mappings. Systems demanding exact mappings could relax some of their constraints, i.e. allow for less restricted mapping logics to take place, to increase query recall for example.

Mapping implementation (MI). How mapping is carried out is an important design issue. Peers could use a local copy of thesauruses such as WordNet, build own dictionaries, construct mapping tables, or when feasible, exchange ontologies (schemas) to translate concepts between ontologies. The choice of the approach to carry out mapping is affected by the scope of the application. For small and domain specific applications, peers could exchange local ontologies or build their own local dictionaries for translation. Larger applications on the other hand, may require local thesauruses which are capable of performing some inference rather than just simple concept-to-concept mappings associated with local dictionaries and tables. Mappings could be carried out automatically, semi-automatically or manually.

Mapping correctness measurement (MC). Correct semantic mapping is fundamental to decentralized semantic knowledge or information sharing. Various research efforts have been devoted to the classification of possible types of faults, measuring the quality of the mapping and estimation of information loss during query propagation and translation. The correctness of mapping is measured in two different ways: numerical and logical measurement. Numerical measurement pertains to the numerical values returned from the mapping operation. For example, a mapping operation could conclude that the semantic relationship between a *Laptop* concept and a *Notebook* concept is equal to 1.0: $(c_1; c_2) = 1.0$, and the semantic relationship between an *Operating system* concept and a *Software* concept is equal to 0.5: $(c_3, c_4) = 0.5$, or some other values. A detailed example related to the numerical values use in the mapping operation could be found in [30]. The logical measurement, on the other hand, is the logical relations that has been concluded by mapping operation. That is, whether or not the relationship between two concepts satisfy the logical operations described earlier in the mapping expressiveness sub-section. The two methods could be modified such that the logical relation could return numerical values and vice versa.

Ownership of mapping (MW). An important decision that SP2P system designers have to make is who (i.e., sender or receiver peer) is going to carry out the mapping. That is, whether query translation takes place before sending the

query or after receiving the query. This is important because it will have an effect on query routing, to the extent that the querying peer will first perform mapping and then submit to only semantically related peers (i.e. if the outcome of mapping is above a certain threshold). This constraint can be used as a strategy for terminating query forwarding. Since the receiving peer performs mappings after receiving a query, this means that any query could be posed to any peer (i.e., there is no restriction on query forwarding). Query receiving peers either answer queries (i.e., if they could translate them to their local representation), or forward them to some other peers.

Mapping Maintenance (MM). Recently, several studies have focused on the mapping maintenance issue and its effects on the SP2P systems reliability [4,10,29,30,31]. These studies have concluded that mapping between different ontologies (schemas) need maintenance. This is because mapping could get corrupted as a result of ontology changes. Corrupted mapping puts the entire system at risk of failure. Hence, there is a need for 1. semantic mapping maintenance, 2. mapping corruption detection, and 3. mapping corruption tolerance. Mapping maintenance is needed to prevent it from corruption, corruption detection is required so it can be fixed, and lastly, mapping corruption tolerance is necessary in order to limit the level of the damage that mapping corruption have done to the system. Figure 5 is a class view of the Mapping construct for the proposed SP2P reference architecture.

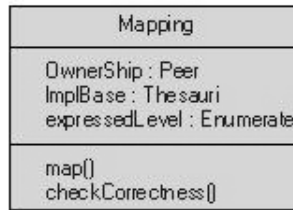


Fig. 5. Mapping construct

Examples: M between concepts and domain representation is fundamental building blocks in Piazza , KEx, Chatty Web systems, and P2PSLN systems. However, the mapping and the implementation of its related issues are different among these systems. For example, while Piazza, KEx and P2PSLN have developed detailed mapping algorithm for their system, Chatty Web relies on reusing existing mappings. In Chatty and P2PSLN the *MW* belongs to querying peer, but in KEx to queried peer. Further, *MC* of mapping is an issue of concern for P2PSLN, and scrutinized deeply in Chatty Web, but others care less about it.

4.6 Router

A Router $T = \{ FS, CH, TT \}$ is an essential component of any SP2P system. The Router component is responsible for delivering query content q from the

query initiator P_i , to one or more query receiver in Neighborhood N. There are three different design aspects relevant to routing queries in SP2P systems. These aspects are: i. Query forwarding strategy FS , ii. Cycle handling CH , and iii. Routing termination policy TT . Existing SP2P systems have different takes on these aspects. Below, each of these issues is described briefly.

Forwarding Strategy (FS). There are several routing strategies for SP2P networks. These include: flooding, random selection of peers, adaptive query routing, sequential routing, etc. These strategies are different from each other, among other things on their usage of number of messages (queries) and time efficiency in retrieving query answers among other. Flooding, for example, is a n^2 query routing algorithm, where n is number of peers in the system. Sequential routing, on the other hand, require only n message, but requires more time to retrieve answers. This is because sequential routing ceases the power of parallelism query routing. The number of messages in other strategies falls between these two extremes, i.e., flooding and sequential routing. **Adaptive query routing** (see e.g. [28] for discussion on adaptive query routing strategies), is the most widely used technique. SP2P systems with adaptive routing strategy utilize learning techniques to enable efficient routing, i.e., peers use their past interaction experience to determine future query routing. In this regard, each peer could consider only its own experience in making decisions on future routing, or in addition to its own experience, it could make use of other peers' recommendation as well. The central idea in adaptive strategy technique is to make usage of the extra information existing in the network to send queries only to the most relevant peers (experts).

Handling Query Repetition (CH). Another important issue of querying SP2P systems is how to deal with query repetitions. Repetitions are commonly identified by using either query unique identifiers (qid) and/or query path information ($path$). A peer may receive the same query from different paths or via a cycle in the network. Alternatively, a peer could receive a more specific query (or a more general one) via different paths or cycles in the network after multiple translations by semantically related peers. The way repeated queries are dealt with has an impact on the number of query message exchanges and result completeness. While terminating already seen queries can preclude the opportunity to provide some important answers, processing repeated queries increases the number of query messages a system would exchange.

Query Termination Policy (TT). When query forwarding is going to stop is another important matter of routing queries in SP2P systems. Current common techniques for stopping query forwarding depend on either counting the number of hops or setting the query time-to-live (TTL). Using the hop-counting approach, a system administrator sets the length of a network path that a query message could traverse before terminating. On the other hand, the TTL approach is time based, i.e. a query message could traverse the network for the period of time that is specified in the query. As a query message traverses the network, its TTL value decreases. When the TTL value becomes zero, message forwarding stops. Note that, these techniques have an impact on the query results that

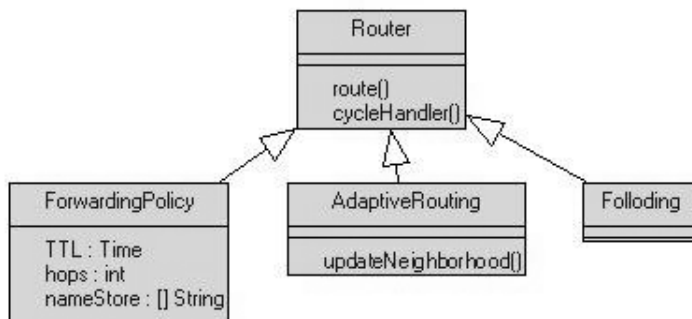


Fig. 6. Router construct

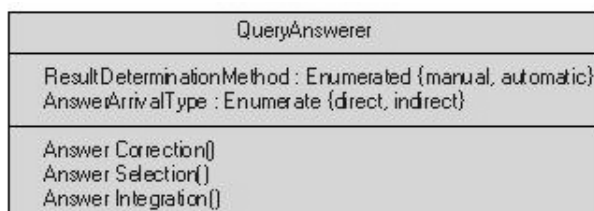


Fig. 7. Query Answerer construct

could be obtained. For instance, peers will continue to forward queries to their related neighbors even when they already have answers to the query as long as the specified constraints permit. As a result the number of query results will be affected. Figure 6 represents a Router Class and its associated forwarding policy.

Examples: T is fundamental building construct of Chatty Web, KEx, P2PSLN and Piazza. Chatty Web, for example, uses adaptive query routing strategy for forwarding queries *FS*. That is, Chatty Web peers use their prior query results to determine which peers they are going to send a query to. They do so by changing the level of confidence that peers have in their out-going links. Chatty Web uses both the TTL and unique query identification as query termination policy *TT*, and to detect cycles *CH*.

4.7 Query Answerer

Query Answerer $QA = \{AE, AD, AS, PAI, AV\}$ concerns with two important aspects of query answers: i. query answer evaluation *AE*, and ii. query answer selection *AS*. Query answers need to be evaluated for their correctness, i.e. correctness (incorrectness) of query answers needs to be determined. For the SP2P systems to be dependable, they need to employ correct result evaluation function. Incorrect evaluation function could prevent semantically related peer from

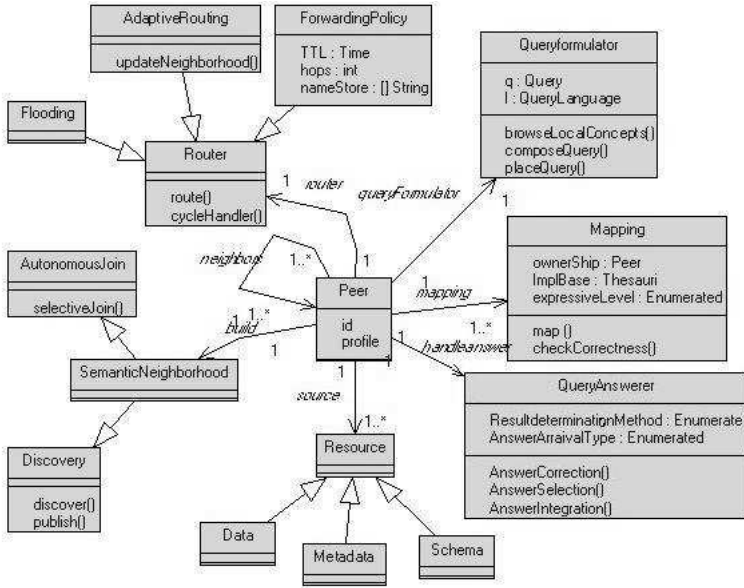


Fig. 8. Semantic Based P2P classes and their dependency

teaming-up together. This in turn, based on the working application, could have far reaching consequence on the performance and dependably of the system. This work will examine this issue in detail, and draw conclusions about the relation between SP2P system reliability and query answer evaluation function.

The way answer evaluation are determined *AD* could be automatic or manual. In manual query answer determination, system users decide on the correctness (incorrectness) of query answer. Automatic query answer determination is about the system peer’s ability to conclude the query answer’s correctness (incorrectness). In the latter case, the system designer needs to design a set of criteria to empower SP2P systems with the ability to decide on the correctness (incorrectness) of query answers. An example of such measurement includes calculating the semantic relation between query answer concepts and query’s concepts. Commonly used answer correctness metrics are precision *AP* and/or recall *AR*.

Answer Selection *AS* { *AP*, *LP*, *W*}, on the other hand, defines a set of criteria for selecting an answer when multiple correct answers generated for a single query – each from a correct translation sequence. This could include answer Precision *AP*, the length of mapping path *LP*, and the level of trust the querying peer has in the peers participating in the result generation, i.e. peer weight *W*.

Another important element of query answer handling is peers’ ability on partial answer integration *PAI*. Some of query results might be partial answers, hence the need for the peer s’ ability to integrate multiple partial answers. That is, Peers need to be capable of combining partial answers and give a uniform view of the results to the users and other peers.

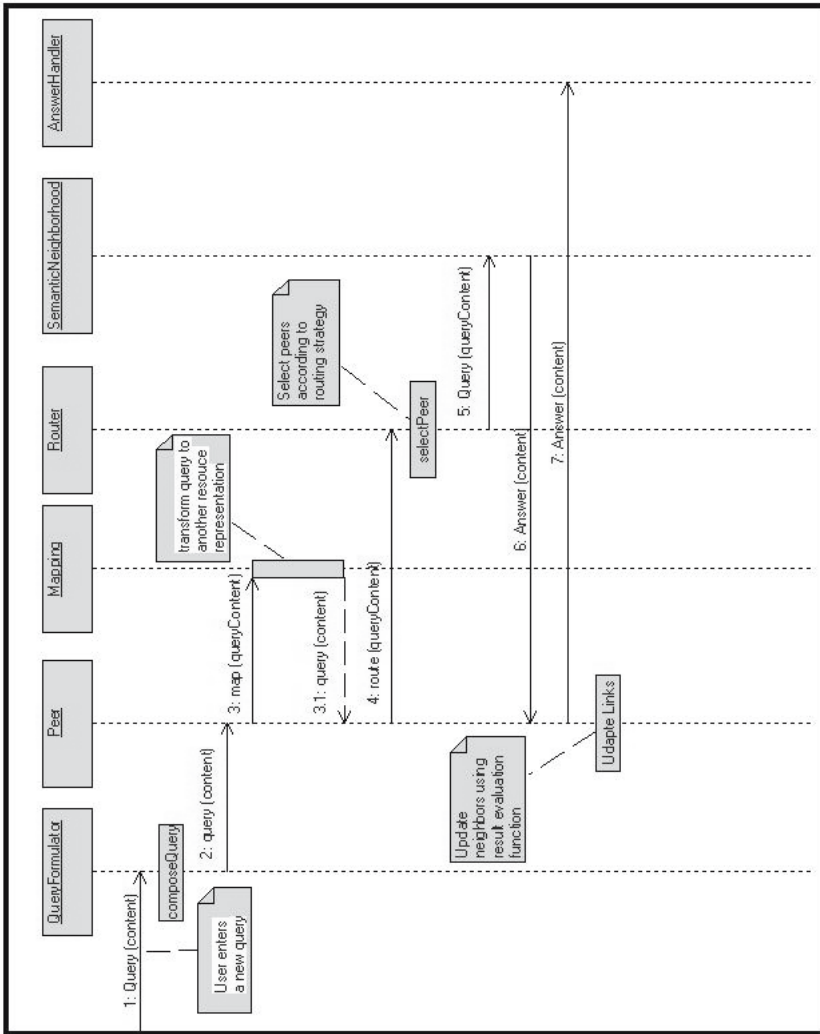


Fig. 9. MSC for SP2P Model Component Interaction

Answers could arrive AV to the querying peer either directly or indirectly. Direct answers are those answers that responding peers provide them directly to the querying peer and without passing through intermediary peers. Indirect answers refer to those that travel along query mapping path to reach the querying peer. Figure 7 is a Query Answerer construct of the proposed SP2P Model.

Examples: Chatty Web puts considerable effort on QA. The querying peer tries to determine automatically if the returned document meets querying peer’s need. P2PSLN system relies on integrating Chatty Web’s QA approach into their

system. Piazza, however, relies on the system user to evaluate query answers, and it has a clear support for PAI.

Figure 8 is put together all model constructs and is the model class diagram and their dependency relationships. Message sequence chart (MSC) showing the interaction between system components is provided in Figure 9.

5 Conclusion and Future Work

In this paper, we have identified that the proposed SP2P solutions for the semantic heterogeneity of information representations problem lack the commonality which makes the SP2P comparison and translation into practical implementations difficult. To overcome the lack of commonality problem in SP2P networks, we have described a reference model for SP2P networks. The model contributes to the advancement of the current SP2P networks in different ways. The minimum necessary constructs to build the SP2P networks and their related design issues have been identified and defined. This empower researchers and network architects to focus on core components and their related design issues, as well as reducing conceptual ambiguity of semantics and meanings of network constructs. The model is also a step toward a standardized API for SP2P networks. Since a particular system can be considered an instance of the reference architecture, simulations build based on the model specification could be used to evaluate different aspects of the existing SP2P networks. For example, how a new routing algorithm, fault-tolerant add-on, etc would affect an existing SP2P networks.

References

1. Aberer, K., Alima, L.O., et al.: The essence of P2P: a reference architecture for overlay networks. In: Proc. Fifth IEEE International Conference on P2P computing, pp. 11–20 (2005)
2. Aberer, K., Cudré-Mauroux, P., et al.: GridVine: Building Internet-Scale Semantic Overlay Networks. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 107–121. Springer, Heidelberg (2004)
3. Aberer, K., Cudre-Mauroux, P., Hauswirth, M.: Start making sense: The Chatty Web approach for global semantic agreements. *Journal of Web Semantics* 1(1), 89–114 (2003)
4. An, Y., Borgida, A., Mylopoulos, J.: Discovery and maintaining Semantic Mappings between XML Schemas and Ontologies. *Journal of computing Science and Engineering* 2(1), 44–73 (2008)
5. Bianchini, D., De Antonellis, V., Melchiori, M., Salvi, D., Bianchini, D.: Peer-to-peer semantic-based web service discovery: state of the art. Technical Report, Dipartimento di Elettronica per l'Automazione Università di (2006)
6. Bonifacio, M., Bouquet, P., et al.: Peer-mediated distributed knowledge management. In: van Elst, L., Dignum, V., Abecker, A. (eds.) AMKM 2003. LNCS, vol. 2926, pp. 31–47. Springer, Heidelberg (2004)
7. Castano, S., Ferrara, A., Montanelli, S.: H-Match: an Algorithm for Dynamically Matching Ontologies in Peer-based Systems. In: The 1st VLDB Int. Workshop on Semantic Web and Databases (SWDB), pp. 231–250 (2003)

8. Castano, S., Montanelli, S.: Enforcing a Semantic Routing Mechanism based on Peer Context Matching. In: Proc. of the 2nd Int. ECAI Workshop on Contexts and Ontologies: Theory, Practice and Applications (2006)
9. Choi, N., Song, I., Han, H.: A survey on ontology mapping. *SIGMOD Rec.* 35(3), 34–41 (2006)
10. Colazzo, D., Sartiani, C.: Mapping Maintenance in XML P2P Databases. In: Bierman, G., Koch, C. (eds.) *DBPL 2005*. LNCS, vol. 3774, pp. 74–89. Springer, Heidelberg (2005)
11. Dabek, F., Brunskill, E., Kaashoek, M.F., Karger, D.: Building peer-to-peer systems with Chord, a distributed lookup service. In: Proc. 8th Wshop. Hot Topics in Operating Syst. (HOTOS-VIII) (May 2001)
12. Dabek, F., Zhao, B., et al.: Towards a Common API for Structured Peer-to-Peer Overlays. In: *IPTPS* (2003)
13. Ehrig, M.: *Ontology alignment: bridging the semantic gap*. Springer, Heidelberg (2007)
14. Euzenat, J., Shvaiko, P.: *Ontology matching*. Springer, Heidelberg (2007)
15. Fergus, P., Mingkhwan, A., Merabti, M., Hanneghan, M.: Distributed emergent semantics in P2P networks. In: Proc. of the Second IASTED International Conference on Information and Knowledge Sharing, pp. 75–82 (2003)
16. Franconi, E., Kuper, G., et al.: Queries and updates in the coDB peer to peer database system. In: Proc. of *VLDB 2004* (2004)
17. Freenet, <http://www.freenetproject.org>
18. Guarino, N.: Formal ontology and information systems. In: Proc. of Formal Ontology in Information Systems, pp. 3–15 (1998)
19. Gruber, T.R.: The Role of Common Ontology in Achieving Sharable, Reusable Knowledge Bases. In: Proc. of the 2nd International Conference on Principles of Knowledge Representation and Reasoning, pp. 601–602 (1991)
20. Gnutella, <http://gnutella.wego.com>
21. Haase, P., Broekstra, J., et al.: Bibster – A Semantics-based Bibliographic Peer-to-Peer System. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) *ISWC 2004*. LNCS, vol. 3298, pp. 122–136. Springer, Heidelberg (2004)
22. Haase, P., Siebes, R., van Harmelen, F.: Peer Selection in Peer-to-Peer Networks with Semantic Topologies. In: Bouzeghoub, M., Goble, C.A., Kashyap, V., Spaccapietra, S. (eds.) *ICSNW 2004*. LNCS, vol. 3226, pp. 108–125. Springer, Heidelberg (2004)
23. Hai, Z., Jie, L., et al.: Query Routing in a Peer-to-Peer Semantic Link Network. *Computational Intelligence* 21(2), 197–216 (2005)
24. Halevy, A., Ives, Z., Mork, P., Tatarinov, I.: Piazza: Mediation and integration infrastructure for semantic web data. In: Proc. of the International World-Wide Web Conference WWW (2003)
25. Joseph, S.: Neurogrid: Semantically Routing Queries in Peer-to-Peer Networks. In: Proc. Intl. Workshop on Peer-to-Peer Computing (2002)
26. Kementsietsidis, A., Arenas, M., Miller, R.: Managing Data Mappings in the Hyperion Project. In: The 19th Intl. Conf. on Data Engineering, pp. 732–734 (2003)
27. Liu, L., Xu, J., et al.: Self-Organization of Autonomous Peers with Human Strategies. In: Proc. of *ICIW 2008*, pp. 348–357 (2008)
28. Löser, A., Staab, S., Tempich, C.: Semantic social overlay networks. *IEEE Journal on Selected Areas in Communications* 25(1), 5–14 (2007)
29. Mawlood-Yunis, A.-R., Weiss, M., Santoro, N.: Fault-Tolerant Emergent Semantics in P2P Networks. In: Cardoso, J., Lytras, M. (eds.) *Semantic Web Engineering in the Knowledge Society*, pp. 161–187. IGI Global (2008)

30. Mawlood-Yunis, A.-R.: Fault-tolerant Semantic Mappings Among Heterogeneous and Distributed Local Ontologies. In: Proc. of 2nd International workshop on Ontologies and Information Systems for the Semantic Web, pp. 31–38 (2008)
31. McCann, R., et al.: Mapping maintenance for data integration systems. In: Proc. of the 31st international conference on VLDB, pp. 1018–1029 (2005)
32. Mena, E., Illarramendi, A., et al.: OBSERVER: an approach for query processing in global information systems based on interpretation across pre-existing ontologies. *Distributed and Parallel Databases* 8(2), 223–271 (2000)
33. Napster, <http://www.napster.com>
34. Ng, W.S., Ooi, B.C., et al.: PeerDB: a P2P-based system for distributed data sharing. In: Proc. of 19th International Conference on Data Engineering, pp. 633–644 (2003)
35. Parashar, M., Member, S., Browns, J.C.: Conceptual and Implementation Models for the Grid. *Proc. of IEEE Journal* 93(3), 653–668 (2005)
36. Rowstron, A., Druschel, P.: Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems. In: Guerraoui, R. (ed.) *Middleware 2001*. LNCS, vol. 2218, pp. 329–350. Springer, Heidelberg (2001)
37. Rousset, M., Chatalic, P., et al.: Somewhere in the Semantic Web. In: Intl. Workshop on Principles and Practice of Semantic Web Reasoning, pp. 84–99 (2006)
38. Schmitz, C., Löser, A.: How to model Semantic Peer-to-Peer Overlays? *GI Jahrestagung* (1), 12–19 (2006)
39. Staab, S., Stuckenschmidt, S.: *Semantic Web and Peer-to-Peer*. Springer, Heidelberg (2006)
40. Zaihrayeu, I.: *Towards Peer-to-Peer Information Management Systems*. Ph.D Dissertation, International Doctorate School in Information and Communication Technologies, DIT - University of Trento (2006)

Author Index

- Adda, Mehdi 169
Adi, Kamel 212
Aïmeur, Esma 25
Amyot, Daniel 65, 267, 290
Archer, Norm 113, 246
Ashoori, Maryam 89
- Behnam, Saeed Ahmadi 65
Benyoucef, Morad 89
Bouzida, Yacine 212
- Cormier, Catherine 184
- Edirisuriya, Ananda 126
El-Boussaidi, Ghizlane 196
Enayat, Hiba 155
Eze, Benjamin 89
- Felty, Amy 53
Forster, Alan J. 65
- Ghanavati, Sepideh 267
Ghose, Tapu Kumar 227
Grimm, Rüdiger 12
- Hage, Hicham 25
Hattak, Ikhlass 212
Hu, Jun 100
- Jaskolka, Jason 252
- Khedri, Ridha 252
Kuziemsky, Craig 279
- Lefebvre, Eric 196
Leshob, Abderrahmane 196
Lévesque, Ghislain 196
Liu, Xia 279
Logrippo, Luigi 212
- Mankovskii, Serge 212
Matwin, Stan 53
- Mawlood-Yunis, Abdul-Rahman 319
McGregor, Wesley 40
Meletiadou, Anastasia 12
Mili, Hafedh 196
Missaoui, Rokia 169
Muegge, Steven 155
Mussbacher, Gunter 290
- Naak, Amine 25
- Park, Dong-Won 77
Perini, Anna 267
Peyton, Liam 65, 89, 100, 141,
267, 279, 306
Pittaway, Jeff 113
Pourshahid, Alireza 290
- Sabri, Khair Eddin 252
Sandoz, Alain 240
Santoro, Nicola 319
Shamsaei, Azalia 65
Siena, Alberto 267
Singh, Kulwinder 77
Stepien, Bernard 53, 141, 306
Susi, Angelo 267
- Tanev, Stoyan 155
Tavasoli, Amir 246
Tran, Thomas T. 1, 184, 227
- Valtchev, Petko 169
van Oorschot, P.C. 233
- Wan, Tao 233
Weiss, Michael 290, 319
- Xiong, Pulei 141
- Zdravkovic, Jelena 126
Zhang, Richong 1