Borworn Papasratorn
Wichian Chutimaskul
Kriengkrai Porkaew
Vajirasak Vanijja (Eds.)

Communications in Computer and Information Science 55

Advances in Information Technology

Third International Conference, IAIT 2009
Bangkok, Thailand, December 2009
Proceedings

Springer

Communications
in Computer and Information Science 55

Borworn Papasratorn   Wichian Chutimaskul
Kriengkrai Porkaew   Vajirasak Vanijja (Eds.)

Advances in Information Technology

Third International Conference, IAIT 2009
Bangkok, Thailand, December 1-5, 2009
Proceedings

Springer

Volume Editors

Borworn Papasratorn
Wichian Chutimaskul
Kriengkrai Porkaew
Vajirasak Vanijja
School of Information Technology
King Mongkut's University of Technology Thonburi
126 Pracha-U-Thit Rd. Bangmod, Thungkru, Bangkok, 10140,Thailand
E-mail: {borworn, wichian, kk, vachee}@sit.kmutt.ac.th

Library of Congress Control Number: 2009938637

CR Subject Classification (1998): D.2, I.2, D.4, H.5, I.7

ISSN 1865-0929
ISBN-10 3-642-10391-X Springer Berlin Heidelberg New York
ISBN-13 978-3-642-10391-9 Springer Berlin Heidelberg New York

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12797482 06/3180 5 4 3 2 1 0

Preface

At the School of Information Technology, KMUTT, we believe that information technology is the most important driver of economy and social development. IT can enable better productivity, as well as helping us to save resources. IT is giving rise to a new round of industrial and business revolution. We now can have products and services that once were believed to be beyond reach. Without IT, it is impossible for people to realize their full potential.

Businesses worldwide are harnessing the power of broadband communication, which will have a profound and constructive impact on the economic, social development, education, and almost all aspects of our life. This new era of unified communication presents us with new challenges. This is why we should work together more closely to enhance the exchange of knowledge related to effective application of broadband communication and IT.

It is my sincere hope that all contributions to the Third International Conference on Advances in Information Technology (IAIT 2009) will increase our understanding of how we can have effectively apply this emerging technology for the benefit of all people all around the world. I hope IAIT 2009 will also lead to more research that can contribute to a better methodology for IT applications in the era of unified communication.

I am very grateful to all our keynotes speakers for coming all the way to Thailand. I would also like to thank everyone who attended IAIT 2009, and helped to make it a success.

September 2009 Borworn Papasratorn

Organization

IAIT 2009 was organized by the School of Information Technology, King Mongkut's University of Technology Thonburi.

Executive Committee

Program Chair	Borworn Papasratorn (Thailand)
Advisory Committee	Susumu Horiguchi (Japan) Roland Traunmuller (Austria)

Program Committee

Organizing Committee	Wichian Chutimaskul (Thailand) Kittichai Lavangnananda (Thailand) Vajirasak Vanijja (Thailand) Kriengkrai Porkaew (Thailand)

Referees

R. Alcock
C. Arpnikanondt
P. Bouvry
J.H. Chan
Y. Chen
V.Chongsuphajaisiddhi
W.Chutimaskul
O. Halabi
C. Hawksley
A.N. Hidayanto
R. Kalayavinai
A. Kamiya
K. Lavangnananda
W. Lertlum
K. Lertwachara
F. Masaru
P. Mongkolnam
C. Nukoolkit
M. Plaisent
K. Porkaew
S. Sagiroglu
U. Silparcha
G. Stylianou
U. Supasitthimathee
V. Vanijja
S. Wangpipatwong
T. Wangpipatwong
N. Waraporn
B. Watanapa
C. Yu
S. Funilkul
A. Ayanso
N. Thongpapanl
T. Neligwa
M. Ostaszewski

Table of Contents

Performance Analysis of BitTorrent and Its Impact on Real-Time Video Applications

Amuda James Abu and Steven Gordon

Sirindhorn International Institute of Technology, Thammasat University,
Pathumthani 12000, Thailand
james@ict.siit.tu.ac.th, steve@siit.tu.ac.th

Abstract. BitTorrent and similar peer-to-peer file sharing applications can represent a large portion of network traffic. Despite the advantages for BitTorrent users, it can unfairly consume access link bandwidth from other user(s) and applications. It can also rapidly fill up buffers at access routers. We have used a detailed model of the BitTorrent protocol to analyze its performance and impact on real-time video traffic. We have shown that increasing the number of BitTorrent clients and/or upload connections can cause a decrease in download rate due to delayed TCP acknowledgements. We also show the effect of access router buffer size on performance: too small reduces BitTorrent's upload rate, too large increases video jitter and delay.

Keywords: BitTorrent, TCP, Peer-to-Peer File Sharing, Real-Time Video.

1 Introduction

BitTorrent [1] is a popular protocol for peer-to-peer (P2P) exchange of data, such as file sharing. The BitTorrent protocol allows a client to download portions of a file from different remote hosts, thereby avoiding dependence on a single server and potentially decreasing the total download time. To ensure there are sufficient remote hosts to download from, BitTorrent requires a downloading client to also upload files.

The popularity and efficiency of P2P file sharing have resulted in performance problems for end-users and Internet Service Providers (ISPs). Estimates of the portion of all traffic contributed by P2P file sharing range from 40% to 70% [2, 3]. This presents problems for end-users because of the way in which the protocols interact with other applications, such as increasing the delay experienced when web browsing. In addition, the large amount of data uploaded by end-users presents challenges for ISP networks, traditionally engineered for a high download/upload ratio.

In this paper we analyze the performance of BitTorrent and investigate its impact on interactive video traffic in an ISP network. BitTorrent aims to maximize the download rate for the end-user. However BitTorrent implements a tit-for-tat strategy to ensure sufficient amount of data is uploaded so that there are enough remote hosts to download from. Therefore, maximizing the upload rate is also important for BitTorrent. For interactive video traffic, delay and jitter should be minimized, while a small number of packet drops can be tolerated.

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 1–10, 2009.

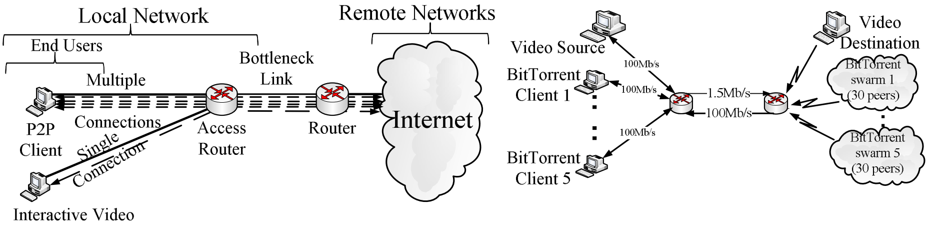
When there are many end-users in an ISP network, the access router that the users connect to, and in particular the uplink from the access router to the next router, may become the bottleneck in the network [4, 5]. BitTorrent, which uploads a large amount of data using multiple TCP connections, may utilize a large portion of that link (and access router buffers), resulting in unacceptable delays for other applications (and non-BitTorrent users). This is particularly detrimental to real-time voice and video applications. The IETF, through the Low Extra Delay Background Transport (LEDBAT) Working Group [6], have also identified these problems for end-users and ISPs, and have recently begun analyzing the issues involved. The results in this paper are a step towards understanding the performance of BitTorrent and its impact on video applications, especially when BitTorrent users are sharing the same access link in an ISP network with a real-time video user. We show the relationship between BitTorrent clients, upload connections, and access router buffer size on delay, upload/download rates and packet drops for BitTorrent and video applications.

This paper is structured as follows: In Section 2, the BitTorrent performance issues are explained. Our assumed scenario is described in Section 3, and the analysis methodology and results are presented in Section 4 Related work is discussed in Section 5, and finally conclusions and future work are given in Section 6.

2 Performance Issues with BitTorrent and Other Applications

BitTorrent is a protocol primarily used for P2P file sharing. In BitTorrent a file is referred to as a *torrent*. The set of *peers* downloading and/or uploading a torrent is called a *swarm*. A peer within a swarm that has fully downloaded the torrent and makes it available to others is a *seed*, while those yet to download the entire torrent are *leeches*. Consider a peer that wants to download a torrent. We will refer to it as the *local peer* and others as *remote peers*. Using a *tracker* server, the local peer discovers a list of remote peers in the swarm, selects N remote peers and establishes TCP connections with each. Then the peers use the *Peer Exchange Protocol* to exchange pieces (i.e. upload and download the file). The strategy for selecting peers to exchange pieces with is important for maximizing the download rate, as well as uploading content for other peers to access. Of the N remotes peers that the local peer is connected to, it will choose U unchoked peers to exchange pieces with. The local peer chooses these unchoked peers from those that it has the best download rates from, and the peers are interested in exchanging pieces with the local peer. Regular updates of choked/unchoked peers are performed, as well as occasional optimistic unchoking in the hope of maximizing the download rate. Further details can be found in the official [1] and unofficial [7] BitTorrent specifications.

Although using multiple TCP connections can increase reliability and distribute traffic load to multiple peers, it can have impact on how link bandwidth is shared among applications and users. In ideal conditions TCP will share the bandwidth of a link equally among the connections. Application X with M TCP connections sharing a link with an application Y with one TCP connection will obtain M times the bandwidth as application Y. Considering an end-user running multiple applications (e.g. BitTorrent, web browsing, file download), the user may experience unfairness in the link bandwidth allocation for each of the applications because of the different number of TCP connections established by each of the applications (e.g. BitTorrent resulting in excessive web response times). However, the end-user has control over this unfairness – the user can manually choose the applications to run, or configure their host to give priority to desired applications. Now consider multiple end-users connecting to an ISP access router as shown in Fig. 1a. If the access router uplink (to the next router) is the bottleneck in the path, then unfairness may arise, this time outside the control of the end-user. Because the access link is a bottleneck, P2P client is capable of rapidly filling up the queue of the access router. Therefore, packets from the video source experience longer queue delay at the access router and they are possibly dropped when the queue is full, especially when the access router uses a drop-tail queue discipline. Large delay and high packet drops can be detrimental to real-time video applications.



(a) (b)

Fig. 1. Showing (a) the sharing of access link among end-users running different applications with different number of connections, and (b) the simulation network topology

Although unfairness between any application with multiple TCP connections and application with single TCP connection may arise (e.g. web browsers use multiple connections), the issue is especially relevant for BitTorrent (and other P2P protocols) for the following reasons:

1. BitTorrent uploads a large amount of data unlike other applications, with the exception of interactive voice/video applications which typically use UDP.
2. BitTorrent local peer may change TCP connections to remote peers on a regular basis (depending on the choking algorithm and availability of peers). This results in changes in TCP parameters (e.g. window sizes) and performance when compared to an application that always uses the same multiple connections for the entire duration of the data transfer.
3. The traffic profile of BitTorrent (data packet sizes, frequency and size of control packets being sent) differs from other applications.

Therefore analysis of the performance of BitTorrent is needed in the access network of an ISP network from the end-user perspective, especially in the presence of other applications used by other end-users in the same access network. We consider the impact of the access router capabilities on BitTorrent under different loads, as well as the interactions with video traffic.

3 Scenario Description and Simulation Setup

We assume an ISP network with *L* local hosts (i.e. customers) all with dedicated links to the ISP access router. The uplink from the access router to the next router is the bottleneck link in the path for end users. This scenario arises when there are a large number of local hosts, each with reasonable uplink speeds, but the uplink speed from access router is insufficient for all hosts uploading at the same time. Beyond the ISP network are *R* remote hosts. To ensure a very high percentage of peer-to-peer traffic in the access network of the ISP network, we assume *L-1* local hosts are running BitTorrent while one local host is running an interactive video application.

The network simulator ns2 [8], with a BitTorrent patch [9], is used to analyze the performance of BitTorrent and its impact on interactive video application using the topology shown in Fig. 1b. The capacity of the access router uplink to the next router is set to 1.5Mb/s while the downlink is set to 100Mb/s. We use these values because we are interested in making the uplink a bottleneck. Other links are set to 100Mb/s in order to congest the access network especially from the local end-users perspective. The delays of all links randomly (uniformly distributed) range from 1-50ms. All routers use drop-tail queue discipline. The default parameter values for the network and applications are shown in Table 1. Queue size of the access router is set high because we want to avoid early packet drop at the access router.

Table 1. Default parameter values

Parameter	Value	Parameter	Value
Local BitTorrent clients	1	*BitTorrent Application*	
Remote peers per swarm	30 per local client	File size	100 MB
Access router queue size	300 pkts	Unchoked connections	4
Link MTU	1500 Bytes	Unchoke interval	10 sec
Video Application		Piece size	256 KB
Data rate	750 kb/s	Block size	16 KB
Packet size	500 B		

TCP New Reno is the transport protocol used by BitTorrent. One of the remote peers is the initial file seed, and each peer remains in the swarm until all other peers have finished the download. This topology (similar to [4]) is chosen to allow the local peer to select from sufficient remote peers and to generate significant traffic on the local link. The video traffic is constant bit rate using UDP. The data rate and packet size are chosen to reflect a good quality video conversation over the 1.5Mb/s uplink.

Two different simulation configurations were carried out: Firstly, BitTorrent traffic with no video session and secondly, BitTorrent traffic with 1 video session in the network. Numerous statistics were collected from the simulations. For brevity, we present the following in this paper: aggregate Uploading and Downloading rate of all local BitTorrent peers; Packet Delay and Packet Drops at access router queue; Inter-arrival time for receiving video client (i.e. an indicator of video jitter). All statistics shown are the average of 10 simulations with different random seeds in each simulation configuration.

4 Analysis and Results

4.1 Number of Local BitTorrent Clients

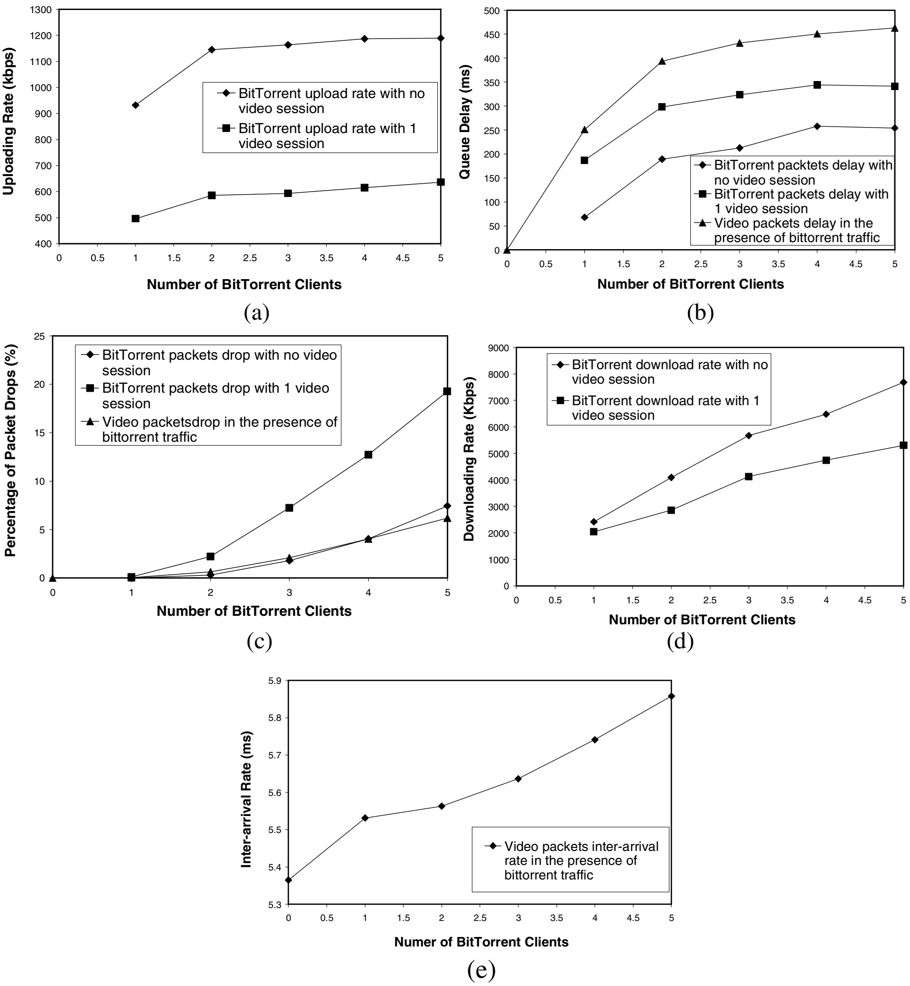
First we consider the impact of varying the number of local BitTorrent clients from 0 up to 5. Results for selected performance metrics with and without the video session are shown in Fig. 2. Fig. 2a shows the aggregate upload rate – the sum of the upload rates for all local peers. With one local peer in the absence of video traffic in the network, the rate is approximately 900kb/s, whereas with two local peers the rate per peer is approximately 575kb/s (a total of 1150kb/s). The aggregate upload rate reaches 1200kb/s. This is limited by the access link rate of 1.5Mb/s (and packet and TCP overheads). However, with a single BitTorrent client, the upload rate is not limited by the access link capacity, but rather by the demand for pieces from the remote peers. In the presence of video traffic, the aggregate BitTorrent upload rate is reduced as the TCP connections across the access router uplink must now compete with the UDP traffic from the video source.

The increasing upload rate of local BitTorrent clients is responsible for an increasing queue delay of packets at the access router. The results in Fig. 2b show that the video packet queue delay at the access router is 0ms with no BitTorrent traffic in the network, but rises sharply as BitTorrent traffic is introduced. Large delay is undesirable for any interactive video application. Large queue delay of BitTorrent packets can affect BitTorrent download rate as TCP ACK packets are delayed.

Arriving packets are frequently dropped when the queue of the access router is quickly filled to its limit (Fig. 2c). BitTorrent packets with 1 video session experience the highest drop rate due to the increasing queue delay and the unavailability of the entire bandwidth of the access router uplink. The reduced drop rate of BitTorrent packets with no video session is due to the availability of the full capacity of the access router uplink. When we have only video traffic present in the network (i.e. 0 BitTorrent clients), no packet is dropped. However, as we introduce BitTorrent traffic, video packets are dropped at the access router. High video packet drop rate is undesirable for the real-time video application. However, BitTorrent packets dropped are retransmitted by TCP, leading to higher load on the access router.

Recall that the downlink rates are effectively unlimited compared to the upload link from access router. The aggregate download rate of local BitTorrent client(s), with and without video session in the network, increases with increasing number of local BitTorrent clients (Fig. 2d). With one local peer the download rate is approximately 2000kb/s, and with two 4000kb/s (2000kb/s per peer). As the local peers are part of independent swarms, it could be expected to see each peer maintaining 2000kb/s download rate. However, this is not the case with 3 or more peers (e.g. with 5 peers less than 8000kb/s). This can be explained by the delayed TCP acknowledgement packets in the queue of the access router. The sending rate of a remote peer (and hence downloading rate of local peer) is limited by the rate at which the remote peer receives TCP ACKs from the local peer.

The video source generates a packet every 5.33ms. Performance is degraded for the interactive video application when the video receiver inter-arrival time increases compared to 5.33ms as the number of local peers increases as shown in Fig. 2e. This is due to the variation of queue delay of video packets at the access router.



(a)

(b)

(c)

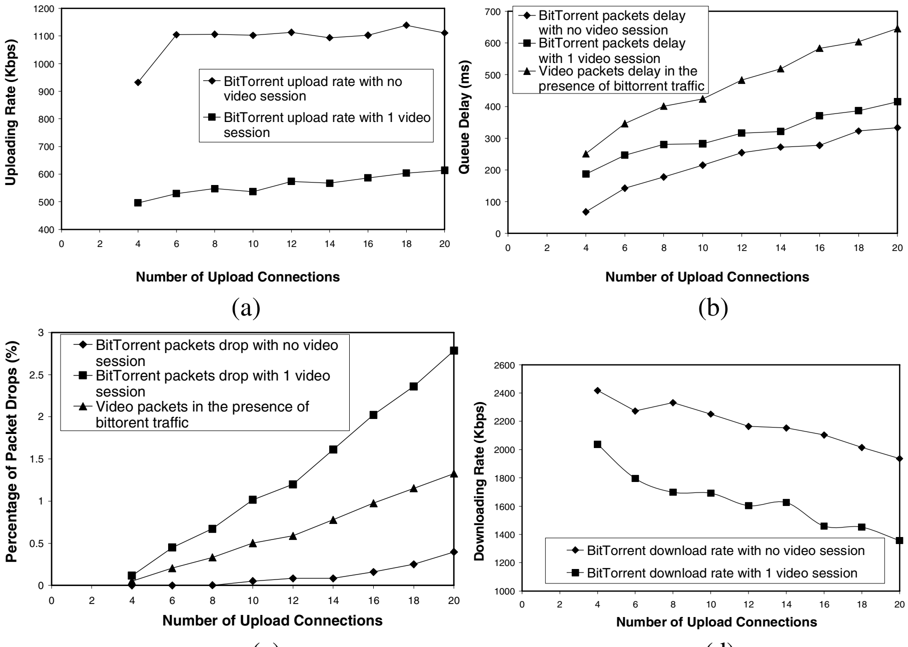
(d)

(e)

Fig. 2. Effects of varying the number of local BitTorrent clients on: (a) BitTorrent upload rate, (b) queue delay of BitTorrent and video packets, (c) percentage of BitTorrent and video packets drops, (d) BitTorrent download rate; with and without video session in the network, and (e) Inter-arrival rate of video packets at the remote host (receiver)

4.2 Number of Unchoked Remote Peers

Now we consider the impact of varying the number of upload connections (unchoked remote peers) from 4 to 20 with a single local BitTorrent client. As shown in Fig. 3a the uploading rate of the local peer (with 1 and zero video session) increases as a result of an increase in the number of multiple TCP connections used by the local peer to upload data to remote peers. An increased number of upload connections of the local peer implies an increased number of remote peers to download from at the same time. As a result, large portion of the file will have been downloaded within a short period of time which can also be uploaded to other interested remote peers.



(a) (b)

(c) (d)

Fig. 3. Effects of varying the number of upload/unchoked connections of a single local BitTorrent client on: (a) BitTorrent upload rate, (b) queue delay of BitTorrent and video packets, (c) percentage of BitTorrent and video packets drops, and (d) BitTorrent download rate; with and without video session in the network

The increasing upload rate is responsible for an increasing queue delay of packets at the access router (see Fig. 3b), with video packets experiencing longer delay which cannot be tolerated by real-time video applications, while coexisting with BitTorrent traffic in the network.

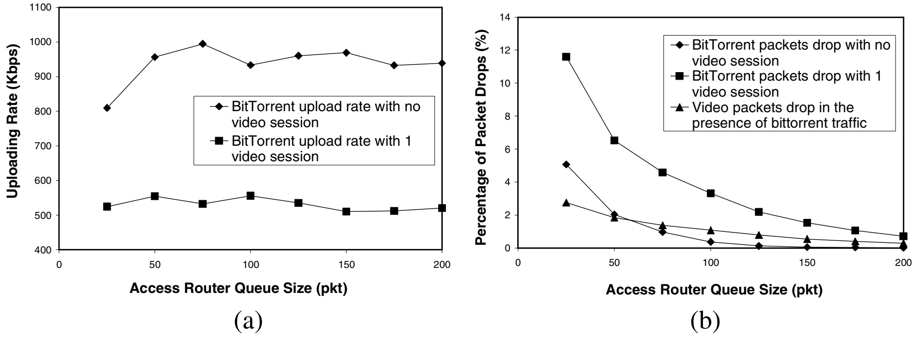
As the queue of the access router grows, it will be eventually filled up to the size limit. Therefore, arriving packets are dropped (Fig. 3c). This is responsible for the increasing percentage of packet drops at the access router. BitTorrent packets dropped are retransmitted by TCP. However, as video packets dropped become large, noticeable portion of the video packets become unavailable at the receiver.

The decreasing download rate for the local BitTorrent client (note that this is for a single client, compared to Fig. 3d which shows the aggregate for all clients) is due to the delayed TCP ACKs in the queue of the access router. 400ms and 300ms queue delays of BitTorrent packets (ACKs) are responsible for the minimum download rates of 1400kb/s and 2000kb/s with and without video session respectively with 20 unchoked connections. This is because the sending rates of remote peers depend on the rates at which they receive ACK packets from the local peer.

4.3 Access Router Queue Size

Finally we obtain results when we vary the queue size of the access router from 25 to 200 packets with an increment of 25. Fig. 4 shows the effects of increasing the access router's queue size on the upload rate of a local peer as shown in Fig. 4a, and the percentage of BitTorrent and video packet drops as shown in Fig. 4b.

With no video traffic and queue size of 25 packets, the upload rate of BitTorrent is poor as the number of packets dropped is large as shown in Fig. 4b, leading to TCP retransmissions with each TCP connection, initiated by BitTorrent, halving its congestion window in response to packet loss, and low uploading rate. However, as queue size increases, the upload rate becomes better as less packets are dropped as shown in Fig. 4b and less retransmission. With video traffic, the Bittorrent upload rate becomes poorer as more packets are dropped. Large video packet drops can be unfriendly for the interactive video application. Low latency tolerance applications such as real-time video perform undesirably with increasing queue size (delay) in the presence of BitTorrent traffic.



(a) (b)

Fig. 4. Effects of access router queue size on: (a) BitTorrent upload rate and (b) percentage of BitTorrent and video packets drops; with and without video session in the network

5 Related Work

Researchers have studied the traffic characteristics of actual P2P file sharing systems, mostly via trace analysis and experimentation, observing factors such as load generated on networks, distribution of pieces among peers, and distribution and activity of peers (e.g. [10, 11, 12]). Such studies provide high-level statistics on BitTorrent traffic across one or more swarms, but do not give insight into the impact of parameters on individual users and ISP networks.

The issues of multiple connections (i.e. parallel downloads) have been studied in the context of P2P file sharing [13] and other applications (e.g. web browsing [14, 15]). Although many issues are similar for non-P2P applications, our analysis is significant because it uses a detailed model of BitTorrent, including aspects of switching between TCP connections, as well as BitTorrent traffic.

An evaluation of the effects of P2P traffic on UDP, in particular voice, is given in [16]. The analysis focuses on wireless access networks, and presents a comparison of voice/P2P performance when QoS mechanisms are used. Although similar results with P2P affecting UDP are seen as in our paper, the effects of multiple connections and access router properties are not considered. It should also be noted that QoS control is not always possible in ISP networks.

ACCM [17] proposes a modified congestion control algorithm to improve fairness between P2P file sharing and other TCP applications. The results show considerable promise for ACCM, but focus only on TCP interactions. The authors are yet to consider real-time voice or video (UDP) traffic.

The IETF LEDBAT Working Group [6] has begun to review the impacts of P2P file sharing on ISPs and end-users. A qualitative analysis of the advantages and disadvantages of multiple TCP connections has been initiated, as well as discussion of transport protocols and congestion control mechanisms suitable for BitTorrent-like applications that improve fairness for other applications. Two promising techniques are Friendly P2P and Ledbat (from BitTorrent). Friendly P2P [4] is a proposed application-level modification of P2P protocols to provide improved fairness between P2P, FTP, and voice applications. Simulation analysis has shown fairness can be improved in the presence of FTP and UDP applications when a single P2P client with multiple connections is operating. Factors such as multiple clients, different number of connections have not been analyzed.

Ledbat [18] is a congestion control scheme used with some BitTorrent/uTorrent applications. The approach is for the local peer to measure delay, and reduce its sending rate before the access router buffer is full, allowing other applications to obtain a fair share of the access router uplink. Although used in real BitTorrent networks, no results or analysis has been reported.

6 Conclusions

In this paper we have analyzed the performance of the BitTorrent protocol and its impact on interactive video application. The analysis is significant as it is one of the first to model the detailed behavior of the BitTorrent protocol, and considers the effects of the access router capabilities, as well as the number of connecting clients in the access network of an ISP network. We have shown that the number of unchoked connections can contribute more to queuing delay than the number of BitTorrent clients. In addition, although upload rates can be increased, because of delayed TCP acknowledgements, the overall download rate reduces. This leads to an important design tradeoff when selecting the number of connections. Finally, the access router queue size has significant impact on application performance: too small can greatly reduce BitTorrent upload rate due to packet drops which leads to halving the congestion window of each TCP connections, and subsequently retransmission; too large will increase delay and jitter for video applications which can jeopardize the performance of such a real-time interactive application. An ISP may not be able to control the number of unchoked connections used by an end-user running BitTorrent application but it can control the number of end-users that mostly use BitTorrent application. Furthermore, an ISP can control the queue delay of packets at the access router by using routers in the access network whose queue sizes are optimal (i.e. not too large because of large delay and not too small because of high percentage of packet drops). As future work we will compare different congestion control mechanisms, both transport and application level, that can deliver fair and efficient access to all applications and users.

References

1. BitTorrent protocol specification, http://www.bittorrent.org/beps/bep_0003.html
2. Ipoque Internet Study (2008/2009), http://www.ipoque.com
3. The Youtube Effect, http://arstechnica.com/old/content/2007/06/the-youtube-effect-http-traffic-now-eclipses-p2p.ars
4. Liu, Y., Wang, H., Lin, Y., Cheng, S., Simon, G.: Friendly P2P: Application-level congestion control for peer-to-peer applications. In: IEEE GLOBECOM, pp. 1–5 (2008)
5. Akella, A., Seshan, S., Shaikh, A.: An empirical evaluation of wide-area Internet bottlenecks. In: Proc. ACM Conf. Internet Measurement, pp. 101–104 (2003)
6. IETF Low Extra Delay Background Transport Working Group, http://www.ietf.org/dyn/wg/charter/ledbat-charter.html
7. Unofficial BitTorrent specification, http://wiki.theory.org/BitTorrentSpecification
8. ns2, Network Simulator, http://www.isi.edu/nsnam
9. Eger, K., Hosfeld, T., Binzenhofer, A., Kunzmann, G.: Efficient simulation of large-scale P2P networks: Packet-level vs flow-level simulations. In: Proc. Workshop Use of P2P, Grid and Agents for Development of Content Networks, pp. 9–16 (2007)
10. Guo, L., Chen, S., Xiao, Z., Tan, E., Ding, X., Zhang, X.: A performance study of BitTorrent-like peer-to-peer systems. IEEE Journal on Selected Areas in Communications 25(1), 155–169 (2007)
11. Bharambe, R., Herley, C., Padmanabhan, V.N.: Analyzing and improving a BitTorrent networks performance mechanisms. In: Proc. INFOCOM, pp. 1–12 (2006)
12. Izal, M., Urvoy-Keller, G., Biersack, E., Felber, P., Hamra, A.A., Garc'es-Erice, L.: Dissecting BitTorrent: Five months in a torrent's lifetime. In: Proc. Passive & Active Measurement Workshop (2004)
13. Koo, S.G.M., Rosenberg, C., Xu, D.: Analysis of parallel downloading for large file distribution. In: Proc. IEEE Intl. Workshop Future Trends in Distributed Computing Systems, pp. 128–135 (2003)
14. Gkantsidis, C., Ammar, M., Zegura, E.: On the effect of large-scale deployment of parallel downloading. In: Proc. IEEE Workshop Internet Applications, pp. 79–89 (2003)
15. Loukopouos, T., Ahmad, I.: Optimizing download time of embedded multimedia objects for web browsing. IEEE Trans. Parallel Dist. Sys. 15(10), 934–945 (2004)
16. Astuti, D., Kojo, M.: Evaluating the behaviour of peer-to-peer IP traffic in wireless access networks. In: Proc. Intl. Network Conf. (2005)
17. Li, W., Chen, S., Liu, Y., Li, X.: Aggregate congestion control for peer-to-peer file sharing applications. In: Proc. Intl. Conf. Software Engineering, AI, Networking and Parallel/Distributed Computing, pp. 700–705 (2008)
18. Shalunov, S.: LEDBAT Congestion Control Algorithm. IETF Internet Draft, draft-shalunov-ledbat-congestion-00.txt (March 2009) (work in progress)

TOF Depth Camera Based 3D Gesture Interaction System

Yang-Keun Ahn, Young-Choong Park, Kwang-Soon Choi, Woo-Chool Park,
Hae-Moon Seo, and Kwang-Mo Jung

Korea Electronics Technology Institute
ykahn@keti.re.kr

Abstract. Active research is underway on virtual touch screens that complement the physical limitations of conventional touch screens. This paper discusses a virtual touch screen that uses a multi-layer perceptron to recognize and control three-dimensional (3D) depth information from a time of flight (TOF) camera. This system extracts an object's area from the image input and compares it with the trajectory of the object, which is learned in advance, to recognize gestures. The system enables the maneuvering of content in virtual space by utilizing human actions.

Keywords: Gesture Recognition, Natural Interaction, Depth Sensor, Virtual Touch Screen.

1 Introduction

Touch screens are being widely used at present owing to the ease of intuitive control. The touch screen, however, cannot be controlled if the production of a touch sensor is impossible (e.g. for large-sized screens), or if direct contact cannot be made in cases where the screen is located far away. To address these problems and facilitate control of touch screens, wide-ranging research has recently been carried out on virtual touch screens [1-2].

Without any physical contact surface in place, a virtual touch screen generates a virtual screen at a given distance from the camera and recognizes the virtual touch of a physical object on the screen.

Kim Hyung-joon [1] and other researchers have proposed a dual camera-based virtual touch screen, which derives the three-dimensional (3D) position of a hand from images inputted from two fixed cameras and recognizes the touch point. Martin Tosas and Bai Li [2] have implemented a virtual screen using a single webcam and a physical grid; when a hand is situated within the framework of the physical grid, as opposed to a virtual grid, a single webcam tracks the hand and recognizes a touch.

Kim's research, however, implements a touch screen based on mathematical calculation of positioning, and as such it is subject to changes in the screen's position, as the screen is fixed. Tosas and Li fail to materialize a virtual screen, as 3D information is not included. Also, both studies implement limited touch features from among widely-varying human actions.

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 11–17, 2009.

A time of flight (TOF) technology-based camera provides 3D depth image information. The results can be expressed in black-and-white images; a virtual touch screen can be implemented in a simple and efficient manner by employing an image processing technique.

A virtual touch screen, however, fails to take into account various human actions, as it simply realizes touch features based on touch points. Against this backdrop, this paper proposes a virtual screen that applies multi-layer perceptron technology to a virtual touch screen to recognize gestures.

The structure of this paper is as follows: Chapter 2 explains a TOF camera-based virtual touch screen, and Chapter 3 discusses a virtual gesture screen based on the multi-layer perceptron. Test results are presented in Chapter 4; a conclusion and suggested directions for future research are given in Chapter 5.

2 TOF Camera-Based Virtual Touch Screen

A virtual screen can be easily implemented by utilizing the 3D depth information of a TOF camera and an image processing technique.

2.1 Depth Image

A TOF camera emits a laser or LED light. Using a built-in sensor, the time taken for the light molecules to return to the camera after touching the object are then recognized, and the distance from that object is calculated [4-5]. The outputs of this camera are 3D depth images, which represent approximate 3D information.

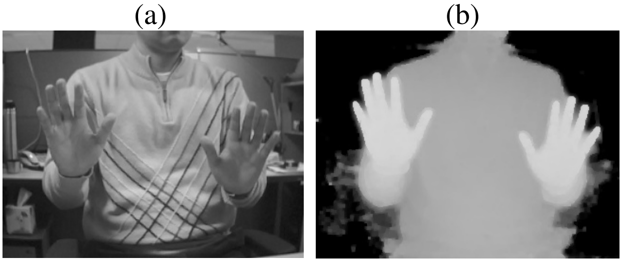


Fig. 1. (a) Image input (b) Depth image

In Fig. 1, (a) refers to the image input of the TOF camera and (b) the corresponding depth map. In (b), the depth map becomes whiter when the object is closer to the camera and blacker when it is farther away.

2.2 Setting of a Threshold Value

A virtual screen can be created by setting a given threshold value for the depth map.

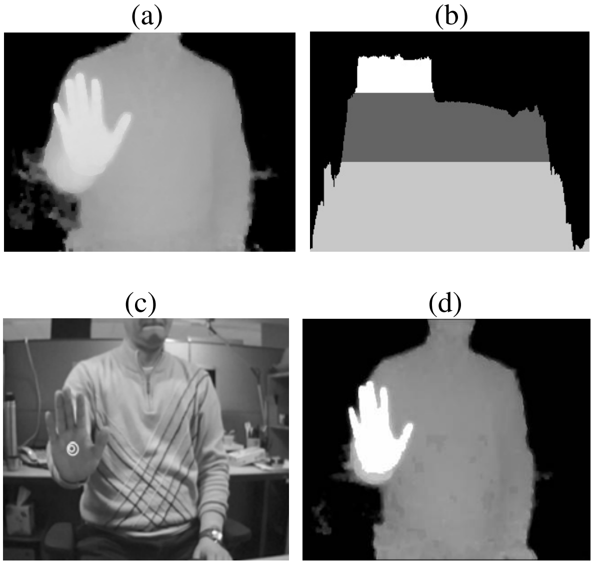


Fig. 2. (a) Depth image (b) X-depth image (c) Object area tracking (d) Virtual screen image

In Fig. 2, (a) is a 3D depth image and (b) an X-depth image, where the depth image is projected onto the X-axis. This X-depth image represents a bird's eye view on the camera. By setting a certain threshold value for the X-depth image, the area of a hand entering the virtual screen can be extracted, as shown in Fig. 2(d). Fig. 2(c) represents an image used in calculating the central moment of the hand area and tracking the hand area that reaches the virtual touch screen. This technique enables the implementation of a virtual screen.

3 Virtual Gesture Screen

A virtual gesture screen can be realized by incorporating an object trajectory database extraction system and a gesture recognition system into the implemented virtual touch screen.

3.1 Object Trajectory Database Extraction System

An object trajectory database extraction system consists of gesture input, gesture image correction, and feature extraction units.

Gesture input. Using a virtual touch screen, touched points are saved to create a gesture database. Whenever a touch action is performed, calculation is done at each frame to determine if the respective frames are touched. If touches are performed on a continued basis, the touched points are saved in the memory, and the gesture is assumed to have ended if an untouched frame is found. Then, as described in Fig. 3, the points are connected to create an image.

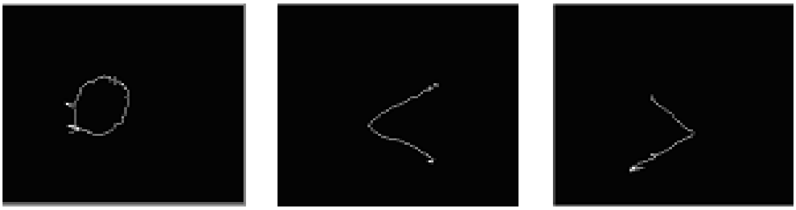


Fig. 3. Gesture input images

Gesture image correction. When the gesture input is over, several phases should be carried out to correct the gesture image.

On this image, even the same gestures may take place in different positions and in differing sizes, and thus individual gestures need to be generalized. In other words, gesture recognition data regardless of position and size are required.

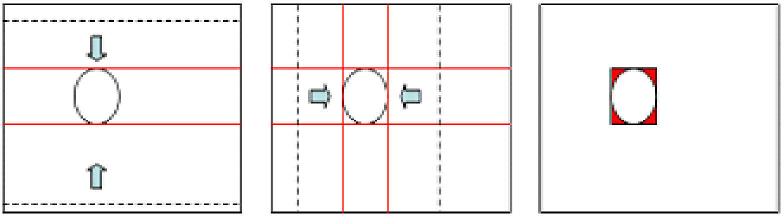


Fig. 4. Positioning of endpoints at the edge of a gesture

For this purpose, the endpoints on the four sides of each gesture image are identified—as shown in Fig. 4—and the image of the area, as described in Fig. 5, is projected onto a 100×100 image. In other words, any image that is smaller than 100 pixels in width or length is enlarged, and one that has width or length greater than 100 pixels is reduced in size.

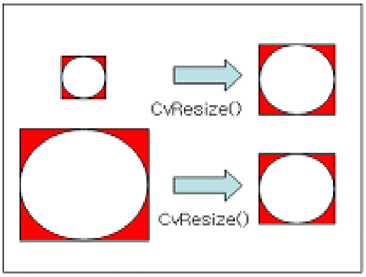


Fig. 5. Readjustment of gesture image size

Feature extraction. After obtaining a gesture image with a size of 100×100 pixels, the features to be used as the input unit in the perceptron should be extracted from the image. This paper divides an image of certain size into 25 smaller images with a size of 20×20 pixels and, as illustrated in Fig. 6, the number of pixels in the gesture part of respective areas is taken as their features. Also, the target value of the perceptron is added to the end of the database. If the total number of gestures is 3 and the inputted gesture is Gesture #1, the Fig. “1 0 0” is entered; if the inputted gesture is Gesture #2, “0 1 0” is added instead. This is because a sigmoid function is used during the learning process as an active function.

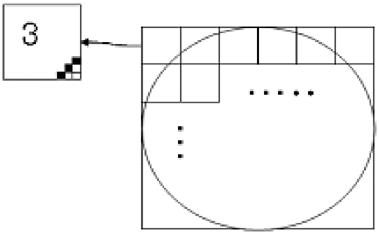


Fig. 6. Number of pixels in the gesture part

3.2 Gesture Recognition System

The gesture recognition system is implemented based on multi-layer perceptron technology.

Multi-layer perceptron. Multi-layer perceptron (MLP) [6-7] is the representative algorithm for supervised learning. The algorithm brings the differences between output values obtained from inputs and predetermined targets of supervised learning (i.e. deviation) back to the perceptron structure and redistributes them in order to gradually reduce deviation levels during the learning process.

Structure of perceptron. The multi-layer perceptron and data from the database explained in Chapter 2 are utilized to train the perceptron. In this case, 25 feature values, specified in 3.1.3, are used as inputs, and the learning target is set at the target value for the database; the number of hidden layers, learning rate, momentum, and target error rate are also determined in advance. Fig. 7 illustrates the structure of the multi-layer perceptron.

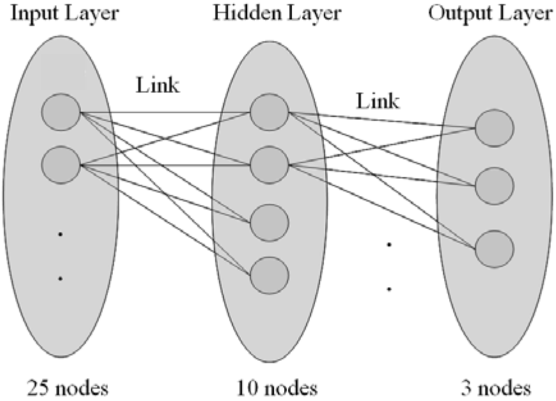


Fig. 7. Structure of multi-layer perceptron

Gesture distinction. After training the multi-layer perceptron, we worked on creating a gesture distinguisher. A real-time trajectory database program is run, and the data produced here are placed into the trained perceptron; its outputs are then used to distinguish gestures. The index of the node with the greatest nodal results for the output layer is used to distinguish a gesture from others. In other words, if the first node of the output layer is greater than the values of the other two nodes, the gesture may be concluded to be Gesture #1.

4 Test Results

To analyze the performance of this system, an experiment is performed to control a flash application with the three gestures described in Fig. 3. The number of hidden layers for the multi-layer perceptron is set at 10, the learning rate at 0.1, and the momentum at 0.8. The target error rate is set to be 0.05 before initiating the learning process; five databases per gesture are used for the experiment.

The experiment shows that the three gestures are different from each other and share no common aspects, and consequently less than 100 iterations of repetitive learning is sufficient to successfully distinguish the gestures.

The flash program provides a menu selection feature when the user makes the first gesture of drawing a circle. The menu is turned to the right when the user draws a right arrow and to the left when the user makes the gesture of drawing a left arrow. Fig. 8 demonstrates how the flash application is controlled using the virtual gesture screen.

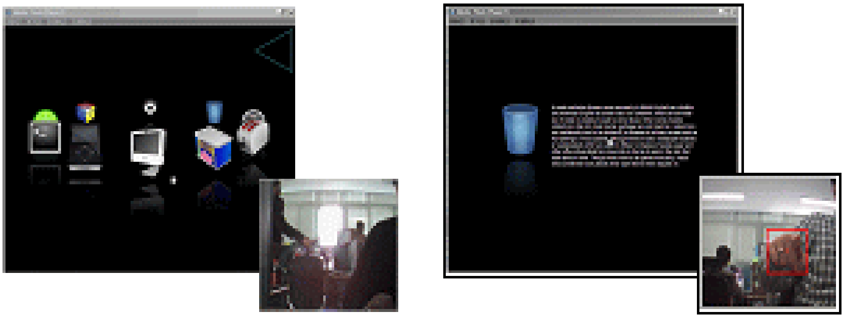


Fig. 8. Controlling of flash program on virtual gesture screen

5 Conclusion

This paper has applied an object trajectory database extraction system and a multi-layer perceptron technology-based gesture recognizer to the conventional virtual screen system to enable gesture recognition as well as virtual touching.

This makes it possible to express wide-ranging human actions, which can be expressed only in a limited manner through virtual touches in 3D space, and to develop a wide variety of content on this basis.

The proposed system can recognize gestures on the virtual screen, but it identifies a single object on a real-time image and is applied only to a single gesture made by that object. The applicability of the system will be further enhanced if a multi-gesture recognition program is developed in the future that enables the recognition of gestures from multiple objects.

Acknowledgements. This research was supported by Ministry of Knowledge Economy(MKE), Korea as a project, "The next generation core technology for Intelligent Information and electronics".

References

1. Hyung-joon, K.: Virtual Touch Screen System for Game Applications. Journal of Korea Game Society 6(3), 77–86 (2006)
2. Tosas, M., Li, B.: LNCS, pp. 48–59. Springer, Heidelberg (2004)
3. Koh, E., Won, J., Bae, C.: Vision-based Virtual Touch Screen Interface. In: Proceeding of ICCE 2008, LasVegas, USA (2008)
4. `http://en.wikipedia.org/wiki/Time-of-flight`
5. Gokturk, S.B., Yalcin, H., Bamji, C.: A Time-Of-Flight Depth Sensor - System Description, Issues and Solutions. In: CVPRW 2004, p. 35 (2004)
6. Hajela, P., Fu, B., Berke, L.: Neural networks in structural analysis and design: an overview. Comput. Syst. Engng. 3(1-4), 525–538 (1992)
7. Lippmann, R.P.: An introduction to computing with neural nets. IEEE Acoust. Speech Signal Process 4(2), 4–22 (1987)

Mining Top-*K* Periodic-Frequent Pattern from Transactional Databases without Support Threshold

Komate Amphawan[1,2,3], Philippe Lenca[2,3], and Athasit Surarerks[1]

[1] Chulalongkorn University, ELITE laboratory, 10330 Bangkok, Thailand
[2] Institut Telecom, Telecom Bretagne, UMR CNRS 3192 Lab-STICC, France
[3] Université européenne de Bretagne
{g48kmp,athasit}@cp.eng.chula.ac.th,
philippe.lenca@telecom-bretagne.eu

Abstract. Temporal periodicity of patterns can be regarded as an important criterion for measuring the interestingness of frequent patterns in several applications. A frequent pattern can be said periodic-frequent if it appears at a regular interval. In this paper, we introduce the problem of mining the top-k periodic frequent patterns i.e. the periodic patterns with the k highest support. An efficient single-pass algorithm using a best-first search strategy without support threshold, called *MTKPP* (Mining Top-K Periodic-frequent Patterns), is proposed. Our experiments show that our proposal is efficient.

1 Introduction

First introduced in [1], frequent pattern mining (also called frequent itemset mining) plays an essential role in many data mining tasks. There are a lot of frequent patterns mining algorithms for a large category of patterns such as association rules [1], correlations [2], sequential patterns [3], dense periodic patterns [4], frequent patterns with maximum length [5], frequent patterns with temporal dependencies [6], etc. Many works have focused on the efficiency of frequent pattern mining by using various techniques such as depth first/breath first search [7], use of trees/other data structures [8], top down/bottom up traversals [9], vertical/horizontal formats [10] and use of constraints [11][12]. Recent surveys may be found in [13] and [14].

However, two main bottlenecks exist: (i) A huge number of patterns are generated and (ii) Most of them are redundant or uninteresting. To tackle these problems, various approaches have been developed.

Frequent-closed pattern mining algorithms have been proposed to reduce redundant patterns [15] and to mine a compact set of frequent patterns which cover all frequent patterns [16]. A recent survey may be found in [17]. While the previous approaches work at the algorithmic level, another strategy is to rank patterns in a post-algorithmic phase with objective measures of interest [18]. A large number of interestingness measures have been proposed. Interesting surveys and comparisons may be found in [19][20] and [21]. At both levels,

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 18–29, 2009.

constraint-based patterns mining, pushing the constraints using objective measures deeply into the patterns mining process is a very interesting approach [11][12]. This approach uses efficient pruning strategies to discover interesting patterns such as optimal rule mining [22][23]. It is important to notice that most of the previous mentioned works, except mainly [22] and [23], are always subject to the dictatorship of support for the frequent pattern mining step. Avoiding the use of the support has been recognized as a major challenge, such as mining high confidence association without support pruning [24][25][26], and mining rules without support threshold [27][28].

Top-k frequent patterns mining techniques that allow the user to control the number of patterns to be discovered without any support threshold have been proposed in [29]. Recently, [30] proposed a pattern mining approach with a periodic constraint on patterns appearance and a minimum support constraint. As pointed out by the authors, there are several domains to apply periodic-frequent patterns mining: in a retail market, in web site design or web administration, in genetic data analysis, in stock market, etc. Thus the occurrence periodicity plays an important role in discovering interesting frequent patterns in such applications [4][31].

We here focus on these two bottlenecks and propose a new algorithm to discover the top-k periodic-frequent patterns without support threshold. The remainder of this paper is organized as follows: in Section 2 and 3, the problem of mining periodic-frequent patterns and the top-k periodic-frequent patterns are introduced. An efficient single-pass algorithm, named *MTKPP*, is presented in detail in Section 4. Section 5 reports the experimental study. Finally, we conclude this paper in Section 6.

2 From Periodic Patterns to Top-*K* Periodic Patterns

Tanbeer et al. [30] define a periodic-frequent pattern as a frequent pattern that appears in a database at a regular period (or interval). They define a new periodicity measure for a pattern using the maximum interval at which the same pattern occurs in a database. In addition, they also consider the occurrence frequency. Unfortunately, the traditional frequent patterns mining techniques fail to discover such periodic-frequent patterns because they are only concerned with the occurrence frequency. The authors further propose an efficient tree-based structure, called PF-tree (Periodic-Frequent pattern tree) that enables a pattern growth mining technique to generate the complete set of periodic-frequent patterns in a database [30]. However the user is still in charge of defining the periodicity and the support thresholds.

It is well-known that setting a minimum support threshold is a difficult task for the user. If the threshold is set too small, a large number of patterns will be found which not only consumes more time and space resources, but also burden the users with analyzing the mining results. On the contrary, if the threshold is set too large, there will be very few frequent patterns. This implies that some interesting patterns are hidden because of improper determination of support threshold. That is why many works try to avoid this task, as we mentioned in the introduction.

We here extend the work of [30] and propose a new kind of pattern, namely the top-k periodic-frequent patterns to be discovered in a transactional database. Moreover, we propose an algorithm that discovers the top-k periodic patterns with the highest values of frequency. Our approach has two major advantages. Firstly, it does not need a minimum support threshold. Secondly, our algorithm needs to read the database only once.

3 Problem Definition

We here give some definitions for top-k periodic-frequent pattern mining following the definitions by [30]. We mainly introduce a precise definition of consecutive transaction-ids. This allows us to define an unambiguous definition of the period of a pattern.

Let $I = \{i_1, i_2, \ldots, i_n\}$ be a set of $n \geq 1$ literals, called items. A set $X = \{i_j, \ldots, i_k\} \subseteq I$, $1 \leq i_j < i_k \leq n$ is called an itemset (a pattern).

A transaction $t = (tid, Y)$ is a tuple where tid represents a transaction-id and $Y \subseteq I$ is an itemset. A transactional database TDB over I is a set of transactions $T = \{t_1, \ldots, t_m\}$, where $m = |TDB|$ is the total number of transactions in TDB. If $X \subseteq Y$, it is said that t contains X or X occurs in t and such transaction-id is denoted as t_j^X, $j \in [1, m]$. Therefore, $T^X = \{t_j^X, \ldots, t_k^X\}$, $j, k \in [1, m]$ and $j < k$ is the set of all ordered transaction-ids (*tids list*) where X occurs in TDB.

Definition 1 (Consecutive *tids* of pattern X). *Let t_j^X and t_k^X, be two tids where X appears, $1 \leq j < k \leq m-1$ and such that there is no transaction t_i that contains X with $j < i < k$. Transactions t_j^X and t_k^X, are then defined as consecutive tids of X.*

Definition 2 (A period of pattern X). *We define a period of the pattern X, denoted as p^X, as the number of transactions between two consecutive tids t_j^X and t_k^X of X:*

$$p^X = t_k^X - t_j^X$$

For simplicity reason we will consider that the first transaction and the last transaction (say, t_f and t_l) in TDB are respectively identified as "null" (i.e., $t_f = 0$) and t_m (i.e., $t_l = t_m$) as in [30]. For instance, from Table 1 the set of transactions where pattern ab appears is $T^{ab} = \{1, 4, 6, 7, 8, 11, 12\}$. Therefore, the periods for this pattern are $1(= 1 - t_f), 3(= 4 - 1), 2(= 6 - 4), 1(= 7 - 6), 1(= 8 - 7), 3(= 11 - 8), 1(= 12 - 11)$ and $0(= t_l - 12)$, where $t_f = 0$ and $t_l = 12$. These periods are then helpful when we consider the behavior of a pattern. In particular, the largest occurrence period of a pattern provide the upper limit of its periodic occurrence characteristic.

Definition 3 (Periodicity of pattern X). *Let T^X be the set of all tids where X occurs and P^X be the set of all periods of $X (P^X = \{p_1^X, ..., p_r^X\})$, where r is the total number of periods in P^X. Then, the periodicity of X can be denoted as $Per(X) = max(p_1^X, \ldots, p_r^X)$.*

For example, in the TDB of Table 1, $P^{ab} = \{1, 3, 2, 1, 1, 3, 1, 0\}$ and $Per(ab) = 3$.

Definition 4 (Support of pattern X). *The number of transactions in a TDB that contains $X(|T^X|)$ is called the support of X and denoted $Sup(X)$.*

For example, the support of pattern ab of Table 1 is $Sup(ab) = |T^{ab}| = 7$.

Definition 5 (Periodic-frequent pattern). *A pattern X is called a periodic-frequent pattern if it satisfies both of the following constraints: (i) its periodicity is no greater than a user-given maximum periodicity: $Per(X) \leq \sigma_p \times |TDB|$ and (ii) its support is no less than a user-given minimum support: $Sup(X) \geq \sigma_s \times |TDB|$ where σ_p and σ_s are expressed in percentage of $|TDB|$.*

Therefore, the periodic-frequent pattern mining problem, given σ_p, σ_s and a TDB, is to discover the complete set of periodic-frequent patterns in TDB having periodicity no greater than $\sigma_p \times |TDB|$ and support no less than $\sigma_s \times |TDB|$.

However, as pointed out before it is quite difficult for users to set a definite support threshold if they have no special knowledge in advance. In addition, in some cases, it is natural for user to specify a simple threshold on the amount of periodic-frequent patterns, say the most 100 frequent patterns with periodicity less than $1,000$ transactions. It is thus of interest to mine the most frequent k periodic patterns over transactional databases without the minimum support threshold requirement.

We thus propose the following definition of top-k periodic-frequent patterns.

Definition 6 (Top-k periodic-frequent patterns). *Let us sort the periodic patterns (i.e. patterns with periodicity no greater than $\sigma_p \times |TDB|$) by descending support values; let S be the support of the k^{th} periodic pattern in the sorted list. A pattern X is called a top-k periodic-frequent pattern if it satisfies the following constraints: (i) its periodicity is no greater than a user-given maximum periodicity : $Per(X) \leq \sigma_p \times |TDB|$ and (ii) its support is no less than the support of k^{th} pattern in the sorted list: $Sup(X) \geq S$ where σ_p is expressed in percentage of $|TDB|$.*

4 *MTKPP*(Mining Top-*K* Periodic-Frequent Patterns)

In this section, we introduce an efficient single-pass algorithm, called *MTKPP* (Mining Top-K Periodic-frequent Patterns), for mining the top-k periodic patterns with the k highest supports from transactional databases.

Our algorithm adopts a best-first search strategy to quickly find periodic patterns with the highest values of support. *MTKPP* consists of two phases: Top-k list initialization phase and Top-k mining phase. Both of them are based on the use of a top-k list as presented below.

Top-k list structure. Top-k list is a linked-list with a hash table which is used to maintain k periodic-frequent patterns with highest supports. As shown in Fig. 1, each entry in a top-k list consists of 4 fields: item or itemset name (I), total support (s^I), periodicity (p^I) and *tids* list where I occurs (T^I).

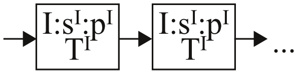


Fig. 1. Top-k list structure

Top-k list initialization phase. To create the top-k list, the database is scanned to obtain all items. At the first occurrence of each item, our algorithm creates a new entry in the top-k list and initializes the support, periodicity and *tids* list. For the other occurrences, *MTKPP* finds the existing entry in the top-k list, using a hash function for efficiency reasons. Then, the values in the entry are updated. After this step, we do not need to scan the database anymore. All items that have periodicity greater than σ_p are removed from the top-k list and the top-k list is sorted in support descending order. Finally, all items that have support less than the support of the k^{th} item in top-k list (s_k) are removed from the top-k list.

Example. Let consider the TDB presented in Table 1. The maximum periodicity threshold σ_p and the number of required results k are 4 and 5 respectively. Figure 2 illustrates the creating of the top-k list from the TDB.

Table 1. Transactional database

tid	items
1	*a b d e*
2	*c d e*
3	*b c f g*
4	*a b d f g*
5	*c e g*
6	*a b c d g*
7	*a b c d*
8	*a b c e*
9	*b c d*
10	*a c e g*
11	*a b f*
12	*a b d g*

With the scan of the first transaction $t_1 = \{a, b, d, e\}$, the entries of the top-k list for items a, b, d and e are initialized as shown in Fig. 2(a). The next transaction ($t_2 = \{c, d, e\}$) initializes a new top-k list entry for item c. It updates the values of support and periodicity for items d and e to $2 : 1$ and *tids* list to $\{1, 2\}$ (Fig. 2(b)). As shown in Fig. 2(c), after scanning the third transaction ($t_3 = \{b, c, f, g\}$), the periodicity p^b of b changes from 1 to 2. The top-k list after scanning all transactions is given in Fig. 2(d). Then, the item f which has the periodicity $p^f = 7$ greater than $\sigma_p = 4$ is removed from the top-k list. Finally, the top-k list is sorted by support descending order and item e is removed from

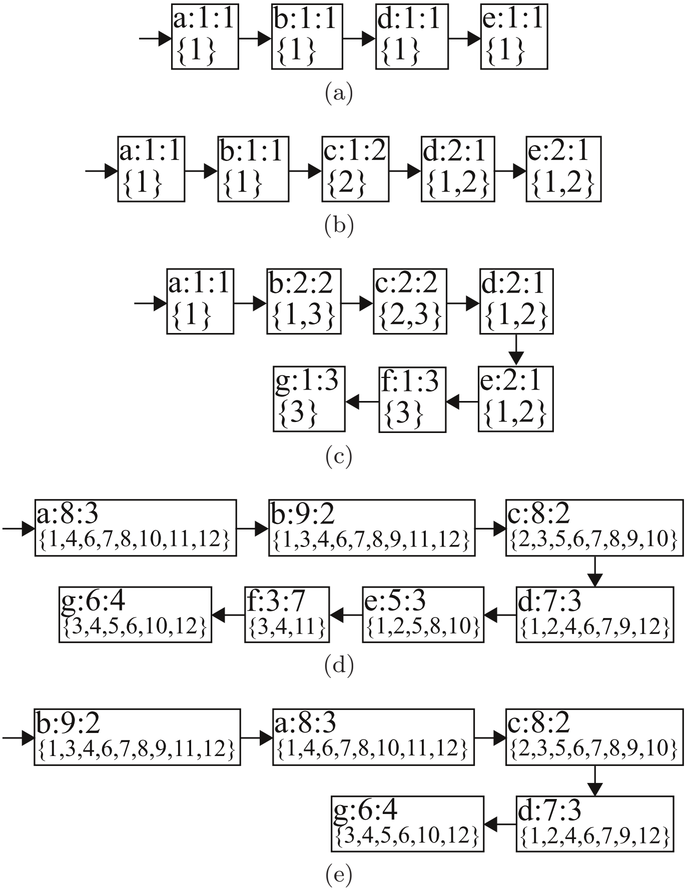


Fig. 2. Top-k list initialization

the top-k list, since the support of $e(s^e = 5)$ is less than support of $g(s^g = 6)$ which is the $k^{th}(5^{th})$ pattern in the top-k list. The top-k list after initialization phase is shown in Fig. 2(e).

Top-k mining phase. To mine all top-k period-frequent patterns from the top-k list, a best-first search strategy is adopted to firstly generate the periodic patterns with the highest support. To generate a new periodic pattern, *MTKPP* starts from considering the most frequent patterns to the least frequent patterns in the top-k list. It then combines two elements in the top-k list under the following two constraints: (i) the size of the patterns of both elements must be equal; (ii) both patterns must have the same prefix (i.e. each item from both patterns is the same, except the last item). When both patterns satisfy the constraints, *MTKPP* will intersect the *tids* lists of the two elements in order to find the periodicity, the support and the *tids* list of the new generated periodic pattern. If the periodicity of the new periodic patterns is no greater than σ_p and the support is greater than the support of the k^{th} pattern in the top-k list, then the newly generated periodic pattern is inserted into the top-k list and the k^{th} pattern will be removed from the top-k list. The details of the mining phase are described in Algorithm 1.

Algorithm 1. (*MTKPP* top-k mining)

```
Input: top-$k$ list, $\sigma_p, k$
Output: top-$k$ periodic-frequent patterns
  for each entry $i$ in top-$k$ list do
    for each entry $j$ in top-$k$ list $(i < j)$ do
      if $|I^i| = |I^j|$ and $|I^i_1| = |I^j_1|, |I^i_2| = |I^j_2|, ..., |I^i_{|I^i|-1}| = |I^j_{|T^j|-1}|$ then
        $p^{i \cup j} = 0, s^{i \cup j} = 0, T^{i \cup j} = \phi$
        for each $tid_x$ in $T^i$ and $tid_y$ in $T^j$ do
          if $tid_x = tid_y$ then
            $s^{i \cup j} = s^{i \cup j} + 1$
            $T^{i \cup j} = T^{i \cup j} \cup tid_x$
            $p = tid_x -$ last tid in $T^{i \cup j}$
            if $p > p^{i \cup j}$ then
              $p^{i \cup j} = p$
              if $p^{i \cup j} > \sigma_p$ then
                stop considering $I^{i \cup j}$ {pattern $i \cup j$ has periodicity $> \sigma_p$}
        if $p^{i \cup j} \leq \sigma_p$ and $s^{i \cup j} \geq s_k$ then
          insert $I^{i \cup j}$ into top-$k$ list
          remove $k^{th}$ entry from top-$k$ list
```

Example. *MTKPP* mine the top-k periodic-frequent patterns from the top-k list of Fig. 2(e). Since item b is the first item in the top-k list and it does not have items in the previous sequence, *MTKPP* starts by considering item a and then looks for other items with the same size and same prefix (which are in the previous sequence in the top-k list), item b. Then, b is combined with a and their *tids* lists are intersected to find the support ($s^{ba} = 7$), the periodicity ($p^{ba} = 3$) and the *tids* list ($T^{ba} = \{1, 4, 6, 7, 8, 11, 12\}$) of pattern ba. Since the periodicity of ba is less than $\sigma_p = 4$ and the support of ba is more than $s_k = 6$, ba is inserted in the top-k list and item g (the k^{th} pattern) is removed from the top-k list (Fig. 3). Next, the third element, item c, is considered. There are two elements which are in the previous sequence and have the same prefix as c: b and a. Then, c is combined with b and their *tids* lists are intersected. The *tids* list and the periodicity of cb are $\{3, 6, 7, 8, 9\}$ and 3 respectively. Because the support of cb ($s^{cb} = 5$) is less than the support of $s_k = 7$, the pattern cb is no longer considered. Next, c and a are combined and their *tids* lists are intersected. The *tids* list of ca is then $\{6, 7, 8, 10\}$. Since the periodicity of $ca(p^{ca} = 6)$ is greater than 4, ca cannot be a periodic pattern. Next, item d and itemset ba are considered in the same manner. When all patterns in the top-k list have been considered, we obtain the top-k periodic-frequent patterns. The final result is shown in Fig 3.

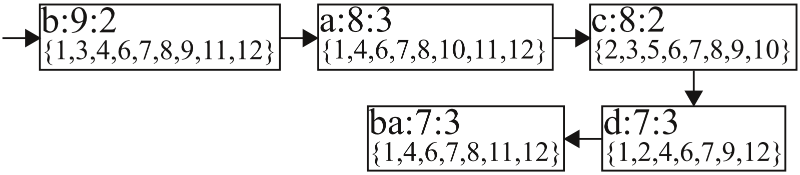


Fig. 3. Top-k frequent patterns

5 Performance Evaluation

In this section, we report our experimental studies to evaluate *MTKPP*. From the best of our knowledge, there is no other existing approach to discover top-k periodic-frequent patterns and we thus only investigate the performances of *MTKPP*.

The *MTKPP* program was implemented in C. The simulations were performed on a 1.6 GHz Intel Xeon with 4 GB main memory on a Linux platform.

5.1 Experimental Setup

We tested our algorithm on one synthetic dataset (*T10I4D100K* [1]) and two real datasets (*Retail* and *Mushroom* [32]).

While *T10I4D100K* and *Retail* are large sparse datasets with $100,000$ and $88,122$ transactions and $1,000$ and $16,469$ distinct items respectively, *Mushroom* is a dense dataset that contains $8,124$ transactions with 119 distinct items.

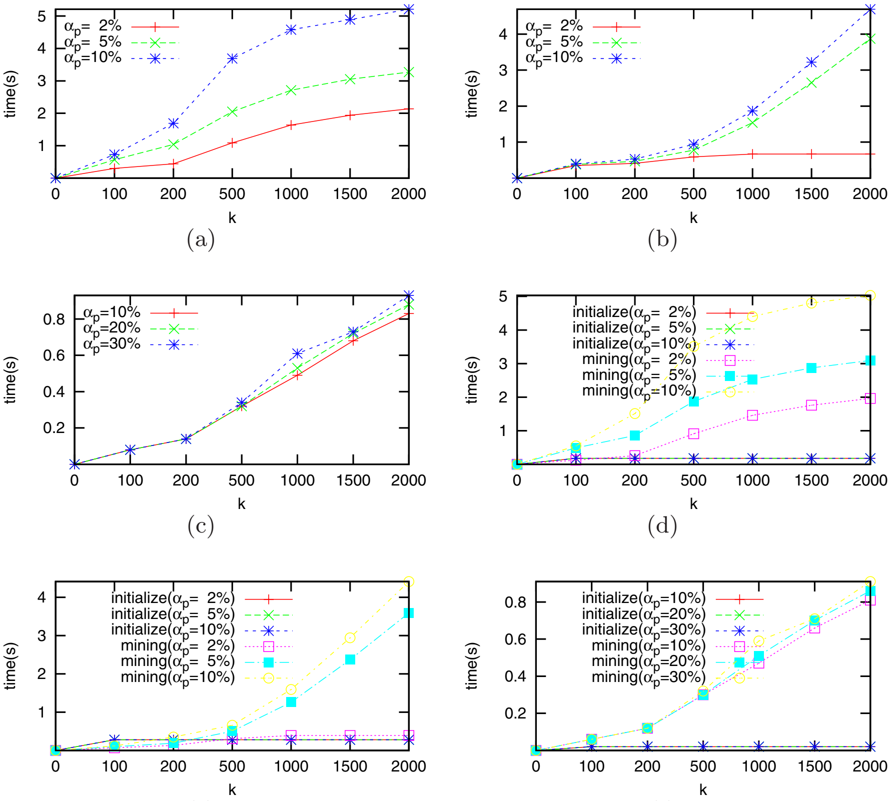
We conducted several experiments to evaluate the performance of our algorithm. We focused on the time and space costs of our approach, where the time cost refers to the time to initialize and mine the top-k periodic-frequent patterns from the top-k list and the space cost refers to the memory requirement of the top-k list.

We evaluate the performance of our algorithm with various values of k and σ_p: from 100 to 2000 for k, and from 2% to 30% for σ_p.

5.2 Execution Time of *MTKPP*

Figures 4(a) and 4(b) give the processing time of *MTKPP* for *T10I4D100K* and *Retail* datasets. One can observe that the computation time increases as k or σ_p increases. When the value of k increases, *MTKPP* has to find more results, therefore the computation time increases as well. When the value of σ_p increases, it causes the increasing of the number of patterns that have periodicity more than σ_p. Thus, the proposed algorithm *MTKPP* has to consider larger patterns as it cannot prune a huge number of patterns only by using the threshold σ_p.

Figure 4(c) shows the computation time of *MTKPP* on the *Mushroom* dataset. One can see that the computational time increases as k increases but it does not increase when the value of σ_p increases. Since *Mushroom* is dense, the patterns



(a) (b)

(c) (d)

(e) (f)

Fig. 4. Computational time of *MTKPP*

in the top-k list occur very frequently and have very low periodicity. Thus, the number of considered patterns is quite stable when the value of σ_p increases.

Figures 4(d), 4(e) and 4(f) give comparisons of the execution time of the top-k list initialization and mining phases. The time to initialize the top-k list is quite unaffected when the values of k and σ_p increase. Indeed, the number of considered transactions and the number of considered items are stable. On the other hand, the time to mine the top-k periodic-frequent patterns increases as k or σ_p increases (Figures 4(a), 4(b) and 4(c)).

5.3 Memory Consumption of *MTKPP*

The variation of memory usage of *MTKPP* with the number of periodic-frequent patterns to be mined, k, is shown in Fig. 5.

From this figure, it is obvious that the memory usage increases as k increases. In fact, the memory usage of *MTKPP* depends on the support of each pattern in the top-k list because *MTKPP* has to keep the *tids* lists of all patterns in the top-k

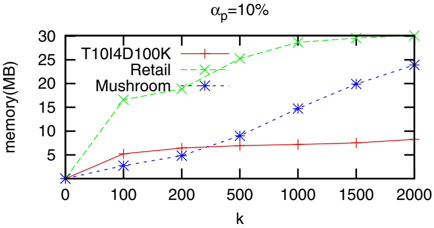


Fig. 5. Memory usage of *MTKPP*

list in order to find the support and the periodicity. For *Mushroom*, the memory usage increases linearly because the supports of patterns in the top-k list do not differ much. For *T10I4D100K* and *Retail*, the memory usage increases slightly as k increases because the supports of patterns in the top-k list are quite different.

6 Conclusion

In this paper, we introduced and studied the problem of mining the top-k periodic-frequent patterns from transactional databases.

An efficient one-pass algorithm, called *MTKPP* (Mining Top-K Periodic-frequent Patterns), is proposed. Since the minimum support to retrieve top-k periodic-frequent patterns cannot be known in advance, a new best-first search strategy is devised to efficiently retrieve the top-k periodic-frequent patterns. It firstly considers the patterns with the highest support and then combines candidates to build the top-k periodic-frequent patterns list.

Our empirical studies, on real and synthetic data, show that our algorithm is efficient and scalable for top-k periodic-frequent pattern mining.

References

1. Agrawal, R., Imielinski, T., Swami, A.N.: Mining association rules between sets of items in large databases. In: Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, D.C., May 26-28, pp. 207–216 (1993)
2. Brin, S., Motwani, R., Silverstein, C.: Beyond market baskets: generalizing association rules to correlations. In: ACM SIGMOD/PODS, pp. 265–276 (1997)
3. Agrawal, R., Srikant, R.: Mining sequential patterns. In: International Conference on Data Engineering, pp. 3–14. IEEE Computer Society, Los Alamitos (1995)
4. Engler, J.: Mining periodic patterns in manufacturing test data. In: International Conference IEEE SoutheastCon., pp. 389–395 (2008)
5. Hu, T., Sung, S.Y., Xiong, H., Fu, Q.: Discovery of maximum length frequent itemsets. Inf. Sci. 178(1), 69–87 (2008)
6. Tatavarty, G., Bhatnagar, R., Young, B.: Discovery of temporal dependencies between frequent patterns in multivariate time series. In: Proceedings of the IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2007, part of the IEEE Symposium Series on Computational Intelligence 2007, Honolulu, Hawaii, USA, April 1-5, pp. 688–696. IEEE, Los Alamitos (2007)

7. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: VLDB 1994, Proceedings of 20th International Conference on Very Large Data Bases, Santiago de Chile, Chile, September 12-15, pp. 487–499 (1994)
8. Han, J., Pei, J., Yin, Y., Mao, R.: Mining frequent patterns without candidate generation: A frequent-pattern tree approach. Data Min. Knowl. Discov. 8(1), 53–87 (2004)
9. Grahne, G., Zhu, J.: Fast algorithms for frequent itemset mining using fp-trees. IEEE Transactions on Knowledge and Data Engineering 17(10), 1347–1362 (2005)
10. Zaki, M.J., Gouda, K.: Fast vertical mining using diffsets. In: Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 24-27, pp. 326–335 (2003)
11. Bonchi, F., Lucchese, C.: Pushing tougher constraints in frequent pattern mining. In: Ho, T.-B., Cheung, D., Liu, H. (eds.) PAKDD 2005. LNCS (LNAI), vol. 3518, pp. 114–124. Springer, Heidelberg (2005)
12. Pei, J., Han, J., Lakshmanan, L.V.S.: Mining frequent item sets with convertible constraints. In: Proceedings of the 17th International Conference on Data Engineering, Heidelberg, Germany, April 2-6, pp. 433–442 (2001)
13. Goethals, B.: Frequent set mining. In: The Data Mining and Knowledge Discovery Handbook, pp. 377–397. Springer, Heidelberg (2005)
14. Han, J., Cheng, H., Xin, D., Yan, X.: Frequent pattern mining: current status and future directions. Data Min. Knowl. Discov. 15(1), 55–86 (2007)
15. Pasquier, N., Bastide, Y., Taouil, R., Lakhal, L.: Discovering frequent closed itemsets for association rules. In: Beeri, C., Bruneman, P. (eds.) ICDT 1999. LNCS, vol. 1540, pp. 398–416. Springer, Heidelberg (1999)
16. Pei, J., Han, J., Mao, R.: Closet: An efficient algorithm for mining frequent closed itemsets. In: ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, pp. 21–30 (2000)
17. Yahia, S.B., Hamrouni, T., Nguifo, E.M.: Frequent closed itemset base algorithms: a thorough structural and analytical survey. SIGKDD Explorations 8(1), 93–104 (2006)
18. Hilderman, R.J., Hamilton, H.J.: Applying objective interestingness measures in data mining systems. In: Zighed, D.A., Komorowski, J., Żytkow, J.M. (eds.) PKDD 2000. LNCS (LNAI), vol. 1910, pp. 432–439. Springer, Heidelberg (2000)
19. Geng, L., Hamilton, H.J.: Interestingness measures for data mining: A survey. ACM Comput. Surv. 38(3), 9 (2006)
20. Lenca, P., Meyer, P., Vaillant, B., Lallich, S.: On selecting interestingness measures for association rules: User oriented description and multiple criteria decision aid. European Journal of Operational Research 184(2), 610–626 (2008)
21. Suzuki, E.: Pitfalls for categorizations of objective interestingness measures for rule discovery. In: Statistical Implicative Analysis, Theory and Applications, vol. 127, pp. 383–395. Springer, Heidelberg (2008)
22. Li, J.: On optimal rule discovery. IEEE Transactions on Knowledge and Data Engineering 18(4), 460–471 (2006)
23. Le Bras, Y., Lenca, P., Lallich, S.: On optimal rule mining: A framework and a necessary and sufficient condition of antimonotonicity. In: Theeramunkong, T., Kijsirikul, B., Cercone, N., Ho, T.-B. (eds.) PAKDD 2009. LNCS, vol. 5476, pp. 705–712. Springer, Heidelberg (2009)
24. Cohen, E., Datar, M., Fujiwara, S., Gionis, A., Indyk, P., Motwani, R., Ullman, J.D., Yang, C.: Finding interesting associations without support pruning. IEEE Transactions on Knowledge and Data Engineering 13(1), 64–78 (2001)

25. Bhattacharyya, R., Bhattacharyya, B.: High confidence association mining without support pruning. In: Ghosh, A., De, R.K., Pal, S.K. (eds.) PReMI 2007. LNCS, vol. 4815, pp. 332–340. Springer, Heidelberg (2007)
26. Le Bras, Y., Lenca, P., Lallich, S.: Mining interesting rules without support requirement: A general universal existential upward closure property. Information Systems (2010)
27. Li, J., Zhang, X., Dong, G., Ramamohanarao, K., Sun, Q.: Efficient mining of high confidence association rules without support thresholds. In: Żytkow, J.M., Rauch, J. (eds.) PKDD 1999. LNCS (LNAI), vol. 1704, pp. 406–411. Springer, Heidelberg (1999)
28. Koh, Y.S.: Mining non-coincidental rules without a user defined support threshold. In: Washio, T., Suzuki, E., Ting, K.M., Inokuchi, A. (eds.) PAKDD 2008. LNCS (LNAI), vol. 5012, pp. 910–915. Springer, Heidelberg (2008)
29. Cheung, Y.L., Fu, A.W.C.: Mining frequent itemsets without support threshold: With and without item constraints. IEEE Transactions on Knowledge and Data Engineering 16(9), 1052–1069 (2004)
30. Tanbeer, S.K., Ahmed, C.F., Jeong, B.S., Lee, Y.K.: Discovering periodic-frequent patterns in transactional databases. In: Theeramunkong, T., Kijsirikul, B., Cercone, N., Ho, T.-B. (eds.) PAKDD 2009. LNCS, vol. 5476, pp. 242–253. Springer, Heidelberg (2009)
31. Laxman, S., Sastry, P.: A survey of temporal data mining. In: Sādhanā, Part 2, vol. 31, pp. 173–198 (2006)
32. Asuncion, A., Newman, D.: UCI machine learning repository (2007)

Intercropping Planning Models for an Agricultural Expert System

Saharat Arreeras, Thara Angskun, and Jitimon Angskun

School of Information Technology, Suranaree University of Technology
111 University Ave., Muang, Nakhonratchasima, Thailand 30000
m5020409@g.sut.ac.th, angskun@sut.ac.th, jitimon@sut.ac.th

Abstract. Traditionally, single cropping culture is widely used. The culture causes more economic risk to growers owing to the crop price is typically fluctuated. Intercropping is an approach of reducing the risk from an economic crisis of crop prices. This paper proposes intercropping planning models for an agricultural expert system. Several models are designed to seek the best approach to the planning by applying a linear programming and many cultivating factors to calculate the maximum income under the concept of economic risk minimization. The experimental results reveal that a model using only a linear programming provided the highest income; but it could not accomplish in the risk minimization. On the other hand, the proposed model could minimize the risk where the related income to linear programming-based model is about 87% on the average.

Keywords: Sustainable Agriculture, Intercropping Planning, Crop Price.

1 Introduction

In the past decades, the agriculture is practiced with the single cropping, i.e., only one of the most profitable crops will be planted at a time. This single cropping causes damages (or changes) to biological diversity and has greater drain on soil nutrients [1]. It also has a high economic risk to growers from surplus of crop. Therefore the challenge is not only emphasized on the biological diversity, but it also considers on the economic risk minimization [2]. Sustainable agriculture can facilitate the mentioned problem. The sustainable agriculture is a method of managing crop ecology in order to maintain the biological diversity, productions and reproductions while it will not harm other ecology systems [3]. One of the sustainable agriculture methodologies is an intercropping, i.e., multiple crops are planted in the same area that conduces to the biological diversity [4, 5]. Although the intercropping is well recognized today, each local area must be able to apply to this theory properly owing to different environments of each area, such as climate, soil fertility and water quantity [6]. In the intercropping planning, many cultivating factors, such as pests, diseases, periods of planting, growing and harvesting, sale prices and cultivating areas, must be considered because those factors are affected to ecological succession and economic risk minimization. Thus, growers have to face with the complication of many factors in order to obtain the optimal plan under the concept of sustainable agriculture. An agriculture expert system is one of information technologies which help to store and

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 30–38, 2009.

manage the cultivation information and automatically generate the cropping plans (e.g., growveg.com [7]). However, these existing expert systems are not considered on the economic risk minimization.

This paper proposes intercropping planning models for an agricultural expert system under the concept of economic risk minimization. The models are designed for a crop group that has been selected according to the biological diversity criteria. The framework of model development for the planning is proposed in section 2. Section 3 describes the model construction. Section 4 presents the model evaluation by comparing between the proposed models and a linear programming-based model. The model enhancement for intercropping planning is discussed in section 5 followed by conclusions and future work in section 6.

2 Model Development Framework

The model development framework for intercropping planning is depicted in Fig. 1 consisting of data, method and process. The linear programming method is a main technique applied to plan the intercropping. There are two types of data used in the model development called user preferences and crop database. User preferences are data defined by users, e.g., total area that users plan to plant all crops. Crop database stores specific details of each crop, e.g., sale prices per plant of each crop, periods of growing until harvesting of each crop and cultivating area per plant of each crop. The data and method are passed through the process to construct, evaluate and enhance the proposed models. In the development, there are 3 major steps of process as follows:

1) *Model Construction* process is to design and create intercropping planning models by applying information as previously described. Several models are generated and passed through the next step in order to test and evaluate.

2) *Model Evaluation* is a process of testing models by comparing results with a linear programming-based model. Then the best model in this evaluation process will be passed through the next step.

3) *Model Enhancement* process uses to improve a performance of the best model. Finally, the model of this process is applied as the intercropping planning model of the agricultural expert system.

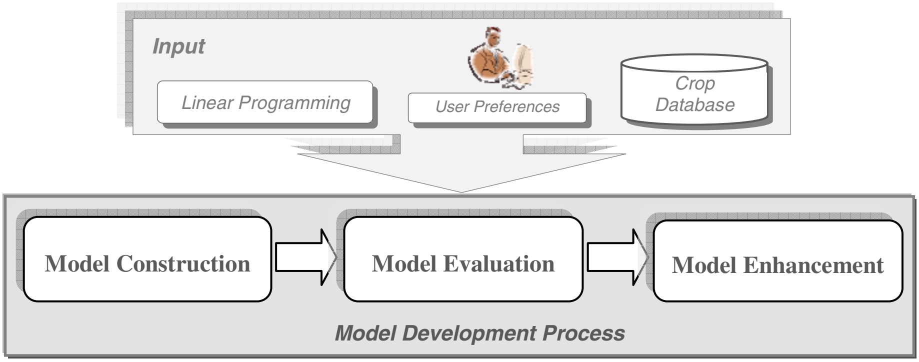


Fig. 1. Model development framework

3 Model Construction

The construction of intercropping planning model uses vegetables in lieu of the crops. The components used in the construction are comprised of three parts as previously described. The first part is user preferences or input data from user, e.g., all cultivating areas that a user requires to plant all selected vegetables or the maximum cultivating area that the user can plant (A). The second part is a database that stores specific details of each vegetable, e.g., sale price per plant of each vegetable (Baht), s_i, where $1 \leq i \leq n$ and n is the number of all types of vegetables; periods of growing until harvesting (Day), d_i and cultivating area per plant of each vegetable (Square meter), a_i [6]. The last part is a technique to solve a planning problem, called Linear Programming (LP) method. The LP, sometimes known as linear optimization, is a method for solving a problem of maximizing or minimizing a linear function [8, 9]. In intercropping planning, the LP method would be applied to find cultivating area distribution that makes total income (I) of all the planted vegetables be maximized where total allocated area (A_{alloc}) of every vegetable is less than or equal to all cultivating areas (A). The results of applying the LP method found that the generated plan was emphasized on maximizing the total income only; but it was not considered on the economic risk minimization (i.e., it still plans with single cropping - almost total areas was allocated to Kale, while allocated areas of Water Spinach and Tomato were near or equal to zero).

The proposed models are constructed to facilitate this problem consisting of four steps as follows.

The first step is to appropriately distribute all cultivating areas to every vegetable before using LP method to maximize the total income. Each vegetable is allocated a specific area as the minimum area that is bounded for individual vegetable. The minimum area of each vegetable ($A_{min}(i)$) is derived from a multiplication of all cultivating areas (A) and a weight of each vegetable (W_i) as the equation 1.

$$A_{min}(i) = A * W_i \, . \tag{1}$$

The weight is generated under the concept of how all the cultivating areas are optimally distributed to individual vegetable. The proposed weight is derived from many cultivating factors which are extensively discussed in the next section.

The second step is finding allocated area of each vegetable. As previously discussed, each vegetable is allocated a specific area; therefore unallocated area after the allocation must be properly assigned to some vegetables. A concept of unallocated area assignment is applying LP method to optimally distribute the areas such that total income is maximized. When the unallocated area ($A_{unalloc}$) is passed through the LP method, the exactly allocated area of each vegetable ($A_{alloc}(i)$) is provided where total allocated area of every vegetable is less than or equal to all cultivating areas (A) as shown in the equation 2, 3 and 4.

$$A_{unalloc} = A - \sum_{\mathrm{i}=1}^{\mathrm{n}} A_{\mathrm{min}}(i) \, . \tag{2}$$

$$A_{min}(i) + LP(A_{unalloc}) = A_{alloc}(i) . \tag{3}$$

$$\textstyle\sum_{i=1}^{n} A_{alloc}(i) \le A . \tag{4}$$

The third step is finding number of cultivated plants of each vegetable (P_i). The number is derived from quotient of the allocated area of each vegetable and the cultivating area per plant of each vegetable (a_i). It is defined by the equation 5.

$$P_i = A_{alloc}(i) / a_i . \tag{5}$$

The final step is to compute the total income (I). The total income is derived from sum of production between the number of cultivated plants of each vegetable and sale price per plant of each vegetable (s_i) as defined by

$$I = \textstyle\sum_{i=1}^{n} P_i s_i . \tag{6}$$

The overall process of model construction is depicted in Fig. 2. The next section discusses about the weight derivation from three cultivating factors: sale prices per plant of each vegetable (s_i); periods of growing until harvesting (d_i); and cultivating areas per plant of each vegetable (a_i).

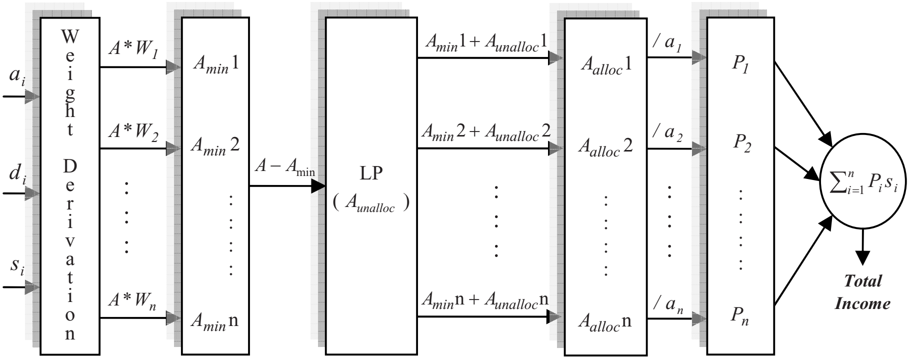


Fig. 2. Model construction process

3.1 Weight Derivation from Sale Prices Per Plant

In this section, weight (W_i) is derived from sale prices per plant of each vegetable (s_i) under an idea that any crop gaining the highest sale price (per plant) should be cultivated in proportion of the most number of plants. However, the weight is used to allocate specific areas for each vegetable and each the vegetable uses cultivating area per plant (a_i) is not equal. Thus an acquisition of proportion of specific area of each vegetable or Weight of each vegetable (W_i) is defined by

$$W_i = \frac{a_i X_i}{\sum_{k=1}^{n}(a_k X_k)} \cdot \qquad (7)$$

Where X_i is a ratio of sale price per plant of each vegetable to sale price per plant of all vegetables. Thus, the weight derivation is summarized as shown in equation 8.

$$W_i = \frac{a_i s_i}{\sum_{k=1,}^{n}(a_k s_k)} \cdot \qquad (8)$$

From the equation, the weights of each vegetable are derived and shown in Table 1.

Table 1. Results of weight derivation from sale prices per plant of each vegetable

Parameters	Chinese kale	Water spinach	Tomato
Sale prices of each vegetable, s_i	2.630	0.370	1.860
Ratio of sale prices per plant of each vegetable, X_i	0.541	0.076	0.383
Proportion of a specific area of each vegetable (weight), W_i	0.295	0.009	0.696

3.2 Weight Derivation from Periods of Growing Until Harvesting

In this section, weight (W_i) is derived from periods of growing until harvesting of each vegetable (d_i) under an idea that any crop having the minimum periods should be cultivated in proportion of the most number of plants because the crop can be produced more number of times per year. Thus the period is inversely proportional to the number of plants (i.e., $X_i \alpha 1/d_i$). In addition, as previously discussed, the weight is used to allocate specific areas for each vegetable and each the vegetable uses cultivating area per plant (a_i) is not equal. Thus an acquisition of proportion of specific area of each vegetable or Weight of each vegetable (W_i) is defined by

$$W_i = \frac{a_i / d_i}{\sum_{k=1}^{n}(a_k / d_k)} \qquad (9)$$

3.3 Weight Derivation from Cultivating Areas Per Plant

In this section, weight (W_i) is derived from cultivating areas per plant of each vegetable (a_i) under an idea that any crop using the minimum cultivating area per plant should be cultivated in proportion of the most number of plants because it could cultivate more number of plants in the same size of area. Thus the area per plant is inversely proportional to the number of plants (i.e., $X_i \alpha 1/a_i$). An acquisition of proportion of specific area of each vegetable or Weight (W_i) is defined in equation 10.

$$W_i = \frac{a_i / a_i}{\sum_{k=1}^{n}(a_k / a_k)} = 1/n \cdot \qquad (10)$$

3.4 Weight Derivation from All the Factors

In this section, weight (W_i) is derived from three factors, i.e., sale price per plant of each vegetable (s_i); periods of growing until harvesting of each vegetable (d_i); and cultivating area per plant of each vegetable (a_i), i.e., $X_i \alpha\, s_i / d_i a_i$. Therefore a proportion of specific area of each vegetable or Weight of each vegetable (W_i) is defined by the equation 11.

$$W_i = \frac{a_i(s_i / d_i a_i)}{\sum_{k=1}^{n}(a_k(s_k / d_k a_k))} = \frac{s_i / d_i}{\sum_{k=1}^{n}(s_k / d_k)} \cdot \qquad (11)$$

4 Model Evaluation

After the weights have been derived, each vegetable is allocated a specific area as the minimum area that is bounded for individual vegetable. The minimum area of each vegetable ($A_{\min}(i)$) is derived as $A_{\min}(i) = A * W_i$.

Table 2. Results of allocated areas of each vegetable and total income

Models	$A_{\min}$			A_{alloc}			Total Income
	Chinese Kale	Water Spinach	Tomato	Chinese Kale	Water Spinach	Tomato	(Baht)
Based on only a LP method	0	0	0	1,599.93	0.06	0	46,754.62
Based on weights derived from prices	472.16	14.76	1,113.08	472.16	14.76	1,113.08	43,672.66
Based on weights derived from periods	454.18	172.59	973.24	454.18	172.59	973.24	40,591.48
Based on weights derived from areas	533.33	533.33	533.33	533.33	533.33	533.33	37,517.85
Based on weights derived from all factors	943.83	227.06	429.11	943.83	227.06	429.11	34,440.85

From a total amount of specific areas of every vegetable, there will be a very small amount of area unallocated. Thus allocated areas for every vegetable (A_{alloc}) is equal to the minimum areas as shown in Table 2. The table is also depicted the total income generated from several models.

The experimental results showed that a model using only a LP provided the most income (46,754.62 baht); but it could not achieve the risk minimization (i.e., almost all areas was allocated to Kale). The other models could minimize the risk; however, the income is dramatically lower than the LP-based model. Therefore the model enhancement must be investigated.

5 Model Enhancement

According to the experimental results, although the model based on weights derived from all cultivating factors gains a high income (approximately 74% related to LP), the model can be improved while it still maintains with the concept of intercropping.

The model enhancement was performed by reducing proportion ($prop$) of minimum area that is bounded to individual vegetable. The new minimum area is derived as the equation 12.

$$new\_A_{\min}(i) = prop * original\_A_{\min}(i) \,. \tag{12}$$

Suppose that the model based on weights derived from all the factors is used to plan intercropping. Thus if $prop$ is 1, new minimum area is equal to the original minimum area as shown in Table 3. If $prop$ is 0.5, new minimum area is equal to one-half of the original minimum area. Then the unallocated area would be distributed to all vegetables optimally using LP method. Finally, total income increases from 34,440.85 to 40,591.48 baht.

Whereas if $prop$ is 0, it means that the model will not specify any area boundary to every crop, i.e., the minimum area of every vegetable is zero. All the cultivating areas are automatically distributed by LP method. Thus the total income of the cultivating factor-based model is equal to that of the LP-based model (46,754.62 Baht) and also the factor-based model is not accomplished in the intercropping concept as the LP-based model.

Table 3. Results of total incomes produced from using different proportions of minimum areas

$prop$	$A_{\min}$			Total Incomes (Baht)
	Chinese Kale	Water Spinach	Tomato	
0	0	0	0	46,754.62
0.5	471.92	113.53	214.55	40,591.48
1	943.83	227.06	429.11	34,440.85

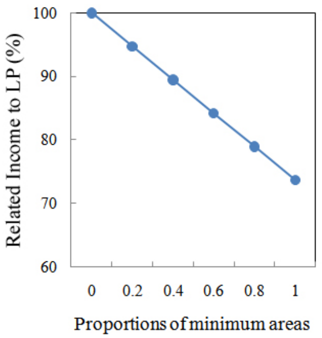


Fig. 3. Comparing incomes of a LP-based model and a proposed model

Fig. 3 shows a comparison of total incomes when derived from the proposed model with several *prop* values (from 0 to 1) related with the LP-based model. The experimental results showed that when *prop* is 0, related income to LP is 100%. While the *prop* is 0.5, related income to LP is 87%. Moreover the related income to LP is at least 74%.

6 Conclusions and Future Work

This paper proposes intercropping planning models for an agricultural expert system. Several models are designed to seek the best approach to the planning by applying a linear programming and many cultivating factors to maximize income while minimize the risk. Several cultivating factors have been investigated and applied to construct the proposed models. These models were evaluated by comparing with a model based on only a linear programming method.

The experimental results reveal that the model using only a linear programming provided the highest income; but it could not accomplish in the risk minimization. On the other hand, the proposed model could minimize the risk where the related income to linear programming-based model is about 87% on the average.

There are several improvements that could be employed in near future such as crop selection. The crop selection is an important and complicated issue to improve the intercropping plan. This is due to the fact that if pests spread over an area, any crop affected by the pests will also be destroyed. Hence the intercropping planning must be considered on common pests, i.e., crops planted in the same area should not have the common pests. The intercropping planning system must have an ability to filter crops whether they cannot be cultivated together, before the system allocates a proper area for individual vegetable.

References

1. German Federal Ministry of Economic Cooperation and Development: Environmental Handbook: Documentation on Monitoring and Evaluating Environmental Impacts: Agriculture, Mining and Energy. Trade and Industry, vol. 2. Friedrich Vieweg and Sohn Verlag (1996)

2. Stuber, C.W., Hancock, J.: Sustaining Plant Breeding-National Workshop. Crop Sci. 48(1), 25–29 (2007)
3. Lewandowski, I., Härdtlein, M., Kaltschmitt, M.: Sustainable Crop Production: Definition and Methodological Approach for Assessing and Implementing Sustainability. Crop Sci. 39(1), 184–193 (2000)
4. Loreau, M., Downing, A., Emmerson, M., Gonzales, A., Hughes, J., Inchausti, P., Joshi, J., Norberg, J., Sala, O.: A New Look at the Relationship between Diversity and Stability. In: Biodiversity and Ecosystem Functioning: Synthesis and perspectives, pp. 79–91. Oxford University Press, Oxford (2002)
5. Picasso, V.D., Brummer, E.C., Liebman, M., Dixon, P.M., Wilsey, B.J.: Crop Species Diversity Affects Productivity and Weed Suppression in Perennial Polycultures under Two Management Strategies. Crop Sci. 48(1), 331–342 (2007)
6. Taylor, N.L.: A Century of Clover Breeding Development in the United States. Crop Sci. 48(1), 1–11 (2007)
7. The Garden Planning Tool, `http://www.growveg.com`
8. Dantzig, G.B.: Springer Series in Operations Research Linear Programming 1: Introduction, p. 1. Springer, New York (1997)
9. Dantzig, G.B.: Linear Programming. Operations Research 50(1), 42–47 (2002)

An Impact of Scheduling Strategy to Parallel FI-Growth Data Mining Algorithm

Nunnapus Benjamas and Putchong Uthayopas

High Performance Computing and Network Center (HPCNC),
Department of Computer Engineering, Kasetsart University, Bangkok 10900, Thailand
{g4885043,pu}@ku.ac.th

Abstract. Parallel computing is very important in providing the computing speed and scalability needed for large scale data mining applications. In order to achieve a good performance, a good scheduling of parallel tasks is very important. This paper proposes and evaluates various scheduling strategies for parallel FI-growth data mining. We show that the execution time of parallel data mining on multicore cluster systems depends on a task scheduling strategy used. Using simulation, we compare 9 strategies on 8 to 64 core multicore cluster systems. The results show that selecting the right strategy can substantially reduce the execution time of parallel data mining on multicore cluster systems.

Keywords: Scheduling strategy, Parallel data mining, FI-growth algorithm.

1 Introduction

The process of Knowledge Discovery & Data mining (KDD) extracts novel significant or interesting knowledge that is implicit in large volumes of data. Association rule mining, an importance component of KDD, resulted from research largely motivated by market basket data analysis. The results allow companies to better understand the purchasing behavior of customers. Association rule mining has been applied to many different domains including inferring patterns from web page access logs, bioinformatics, and the analysis of medical, scientific, and commercial data -- all areas in which relationships between objects can provide useful knowledge.

The process of finding association rules is composed of two phases. First, a set of frequent itemsets from the database should be constructed. Second, the set of frequent itemsets are used to generate “interesting rules”. Most researchers in mining association rules focus on a procedure for finding frequent itemsets, which is the most time-consuming process. When association rule mining algorithms, applications, and tools are implemented on high-performance parallel computers, it opens up the possibility of analyzing much larger databases. Faster processing speed means that users can now experiment with more models to understand more complex data. Therefore, association rule mining applications can benefit from the use of parallel computing systems to improve performance.

In this paper, we propose several scheduling strategies for parallel FI-growth data mining, and compare them using a scheduling simulator. The rest of the paper is organized as follows: Section 2 describes related research in this area. Section 3 describes the parallel FI-growth algorithm and system model. Section 4 proposes the

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 39–47, 2009.

designing the scheduling strategies. Section 5 presents our experimental evaluation. Finally, section 6 presents the conclusion and future work.

2 Related Works

One of a very early algorithm for association rules mining is Apriori [1], [2]. This algorithm base on the generation of candidate itemsets C_k in a pass k using only frequent itemsets L_{k-1} from the previous pass. The idea rests on the fact that any subset of a frequent itemset must be a frequent itemset as well. Hence, C_k can be generated by joining L_{k-1} and deleting those that contain any subsets that are not frequent. Most of the previous studies [3] adopt an Apriori-like candidate set generation-and-test approach. The disadvantages of these algorithms are: (1) candidate itemsets generation requires a very long computation time, (2) the database must be scanned several times, depending on the value of k, so the I/O requirement is rather intensive.

Han et al. [4] presented a new algorithm of mining association rules called the FP-growth algorithm. This algorithm finds a complete set of frequent itemsets by avoiding candidate itemsets generation using a Frequent Pattern tree (FP-tree) structure. FP-tree is a tree structure that contains all prefixes of items in transactions. The basic idea of the pattern growth approach is to grow a pattern from its prefix. Recently, Amphawan and Surarerks [5] presented the Frequent Item growth (FI-growth) algorithm, for creating a new FP-tree called a Frequent Item tree (FI-tree). They have shown that the complete set of frequent itemsets can be generated using a single tree.

Several parallel data mining algorithms have been designed for association rule mining [6] using generation-and-test approach [7], [8]. Those algorithms are based on Apriori [2]. Other research uses a pattern growth approach [9], [10] based on FP-tree. There are three broad strategies for parallelizing association rule mining algorithms [11]. Firstly, task parallelism (or control parallelism) is exploited by having each process executes different operations on the database. Secondly, SPMD parallelism can be employed by having a set of processes executes the same algorithm in parallel on difference partitions of the database [7], [8], [9]. Third, independent parallelism is exploited when processes are executed in parallel in an independent way; generally each process has access to the whole database [7]. Hardware platform issues are also addressed in the parallel implementation of association rule mining, such as shared memory [9] and distributed memory [8], [10]. In the next section, we will briefly describe the concept of parallel FI-growth algorithm and how to increase its performance using various scheduling strategies.

3 Parallel FI-Growth Algorithm

Parallel FI-growth [12] is an algorithm for parallel data mining that parallelizes the association rule mining process. This algorithm employs a data parallelism technique on a multicore cluster (see the workflow in Fig. 1). The workflow consists of 3 phases that is: preprocessing, FI-tree construction, and mining phase. In the preprocessing phase, we first partition the transaction database into several portions, and distribute them to different processors for computation. The FI-tree construction phase, each processor independently constructs its own local FI-tree structure and discovers

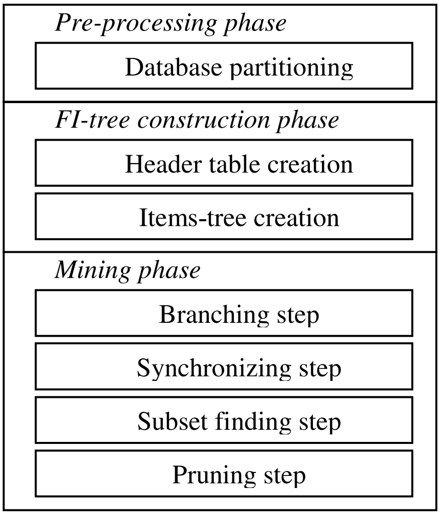


Fig. 1. The parallel FI-growth workflow

corresponding frequent itemsets. However, all processors need to perform a one-time synchronization to exchange their sub-trees before the last two steps in the mining phase.

3.1 Application Model

To study the algorithm, parallel FI-growth program is modeled using a directed acyclic graph (DAG), which is used as input to the scheduling simulator. Given a set

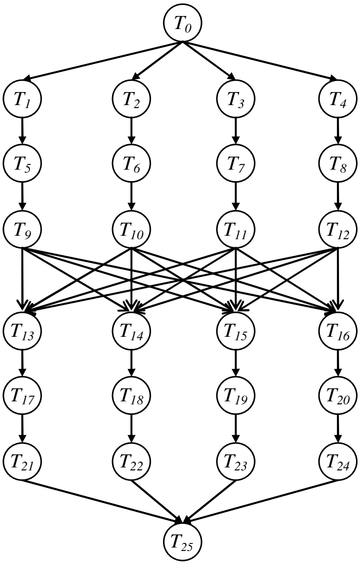


Fig. 2. The parallel FI-growth task graph

of tasks $T = \{T_0, T_1, \ldots, T_n\}$ to be executed, a directed acyclic graph can be defined as $G= (V, E)$, where V is a set of nodes $\{v_1, v_2, \ldots, v_n\}$ and E is the set of directed edges $\{e_{ij}\}$. The edge e_{ij} in the DAG connecting nodes v_i and v_j represents the communication and precedence constraints among the nodes. The weight of an edge denoted by $W(e_{ij})$ is the communication cost of the edge. (See the task graph in Fig. 2). A node v_i in the DAG corresponds to task T_i. Each node of the task graph has a weight $W(v_i)$ that represents the computing cost.

3.2 System Model

In this work, we use a multicore cluster system as the target architecture. A multicore cluster system is a set machine $M = \{m_1, m_2, \ldots, m_{|M|}\}$ of |M| machines and each machine m_i include a set processor $P_i = \{p_1, p_2, \ldots, p_{|pi|}\}$ of $|P_i|$ processor. Each processor p_j of machine m_i has multiple core $C_i = \{c_1, c_2, \ldots, c_{|Cij|}\}$ of $|C_{ij}|$ cores. The Total number of processors is $P_{total} = \sum_{i=1}^{|M|} p_i$ and the total number of cores is

$$C_{total} = \sum_{i=1}^{|M|} \sum_{j=1}^{|p_i|} C_{ij} \cdot$$

In Fig. 3, multicore cluster system is a tree-structure with cores (c) as leaves, processors (p) as intermediate nodes being a parent for cores, machines (m) as intermediate nodes combining processors and the entire machine or system (S) as root node. Let $S = (V, E, W_V, W_E)$, where $V = \{v_1, v_2, \ldots, v_n\}$ is a set of nodes, and E is a set of undirected edges. V represents the set of cores in the system, and an edge $e_{ij} \in E$ represent communication link between core v_i and v_j. $W_V(v_i)$ and $W_E(e_{ij})$ is the computational cost of core v_i and the communication cost between core v_i and v_j, respectively.

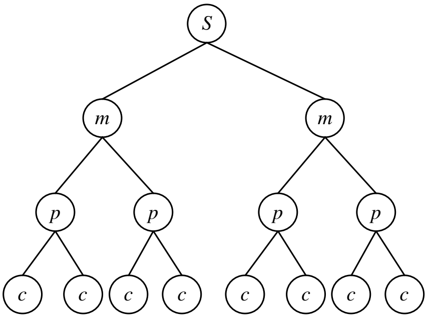


Fig. 3. Machine model

In this paper, a multi-core cluster system will have a hierarchy of communication networks. First, an *inter-core communication* using the shared-cache between cores on the same processor. Second, an *inter-processor communication* using shared-memory between cores on the different processor. Finally, an *inter-machine communication* based on message passing between machines. In this work, we assume that the bandwidth between cores is 20 GBps, bandwidth between processors is 10 GBps, and bandwidth between machines is 1 GBps.

4 Designing the Scheduling Strategies

In order to obtain a good speedup, we propose that the right scheduling should be employed. In order to design the scheduling strategy for parallel FI-growth application, the execution is divided into two steps which is the ordering of the tasks and the mapping of the tasks onto processing units. For each step, the strategy can be formulated as follow.

4.1 Strategies for Ordering the Task

From a directed acyclic graph is used as input to the simulator. We are to make a sequence of task for scheduling by assigning them priority. In this paper, we use 3 ways to determine the priorities of nodes.

Smallest-numbered available task first or Left-to-Right (L-R): Under this priority, a scheduling list from task graph in Figure 2 is T0, T1, T2, T3, T4, T5, T6, T7, T8, T9, T10, T11, T12, T13, T14, T15, T16, T17, T18, T19, T20, T21, T22, T23, T24, T25.

Largest-numbered available vertex first or Right-to-Left (R-L): Under this priority, a scheduling list from task graph in Figure 2 is T0, T4, T3, T2, T1, T8, T7, T6, T5, T12, T11, T10, T9, T16, T15, T14, T13, T20, T19, T18, T17, T24, T23, T22, T21, T25.

Top-to-bottom (T-B): Under this priority, a scheduling list from task graph in Figure 2 is T0, T1, T5, T9, T2, T6, T10, T3, T7, T11, T4, T8, T12, T13, T17, T21, T14, T18, T22, T15, T19, T23, T16, T20, T24, T25.

4.2 Strategies for Task Mapping

We considered three criterions for mapping tasks to processing unit.

Earliest start-time: Allocate the task to a processing unit (core) which allows the earliest start-time first.

Largest size of data transfer: Tasks are allocated by considering the size of data transfer from its parent nodes. Allocate the task to a same processing unit of its parent node which has largest size of data transfer.

Largest size of data transfer & earliest start-time: Tasks are allocated by considering the factors of both the start-time of processor unit and size of data transfer from its parent nodes. At first we consider size of data transfer from its parent nodes and then earliest start-time.

By combining these variations of ordering and mapping strategies, we obtain 9 basic scheduling strategies as shown in Table 1.

Table 1. Possible scheduling strategies

No.	Ordering	Mapping
1	L-R	Earliest start-time
2	L-R	Largest size of data transfer
3	L-R	Largest size of data transfer & earliest start-time
4	R-L	Earliest start-time
5	R-L	Largest size of data transfer
6	R-L	Largest size of data transfer & earliest start-time
7	T-B	Earliest start-time
8	T-B	Largest size of data transfer
9	T-B	Largest size of data transfer & earliest start-time

After formulating these strategies, the next step is to test which strategies will perform the best for our target system which is the large multicore cluster.

5 Experimental Evaluation

In order to generate the test task graph, we mined a 60 million transaction database using the parallel FI-growth program [12]. We utilized the standard "IBM synthetic data generator" [13] to synthesize a transaction database. We used 1000 unique items to create 60 million records; each has average transaction length of 10. After the test graph is generated, we can run a scheduling simulator on this task graph using different scheduling strategies. To evaluate these scheduling strategies, we ran all of the considered strategies on variation architecture as shown in Table 2.

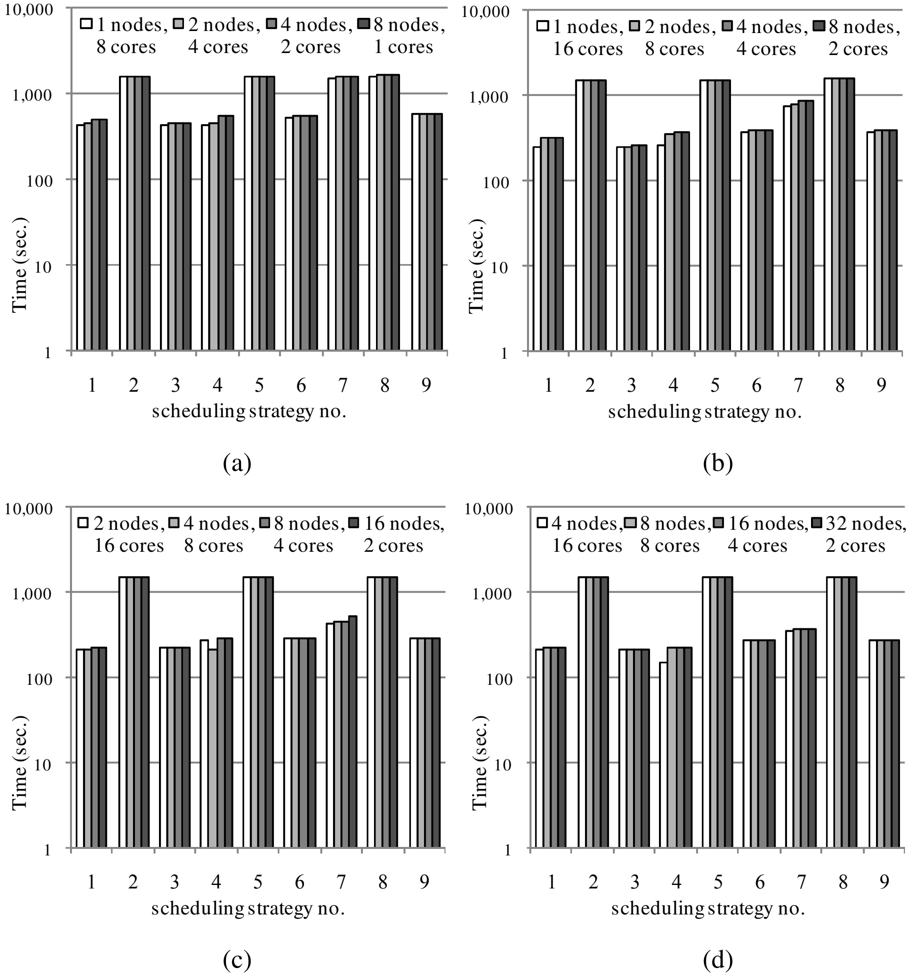
All the scheduling strategies in our simulator have been written in Java programming language. All the experiments are conducted on PC with Core 2 Duo 1.66 GHZ CPU, 2.00 GB memory and hard disk 140 GB. The operating system used is Windows Vista.

Table 2. Multi-core variation architecture

Total number of cores	# of machines	# of per machines
64	4	16
	8	8
	16	4
	32	2
32	2	16
	4	8
	8	4
	16	2
16	1	16
	2	8
	4	4
	8	2
8	1	8
	2	4
	4	2
	8	1

Fig. 4a, 4b, 4c, and 4d illustrate the run times for nine scheduling strategies using 8, 16, 32, and 64 cores, respectively. The results show that run time of *Earliest start-time* strategy (scheduling strategies no. 1 and 4) and *Largest size of data transfer & Earliest start-time* strategy (scheduling strategies no. 3, 6, and 9) is lower than *Largest size of data transfer* strategy (scheduling strategies no. 2,5, and 8). Except run time of *Earliest start-time* strategy that makes a sequence by assigning *Top-to-bottom* priority is more than other scheduling strategies when decrease the amount of cores. The result is shown in Fig. 4d. The execution time of best case is ten times less then worst case.

For the ordering, the run time of *L-R* strategy (scheduling strategies no. 1, 2, and 3 and *R-L* strategy (scheduling strategies no. 4, 5, and 6) is similar and less than *T-B* strategy (scheduling strategies no. 7, 8, and 9) especially when decrease amount of core.



(a)

(b)

(c)

(d)

Fig. 4. Run time of parallel FI-growth of the scheduling variation strategies using (a) 8 cores, (b) 16 cores, (c) 32 cores, and (d) 64 cores on variation architectures

For multi-core cluster system, load balancing among cores becomes a critical issue for high performance. Fig. 5 shows processor workload on 32 core architecture that consists of 2 machines, 16 cores per machine. We can see that the scheduling strategy no. 1, 4, and 7 uses processor workload as the criterion of mapping tasks to cores. By contrast, the scheduling strategy no. 2, 5 and 8 tries to minimize the communication costs between cores. As a result, the schedules generated by scheduling strategy no. 1, 4, and 7 are well load balanced.

In conclusion, scheduling strategies no.1 take time less than the other scheduling strategies and are well load balanced on small, medium, and large cluster configurations. For the architecture of cluster, that have total number of cores be equal but have different number of machines and number of core per machine, has a little affect when work with the same scheduling strategies.

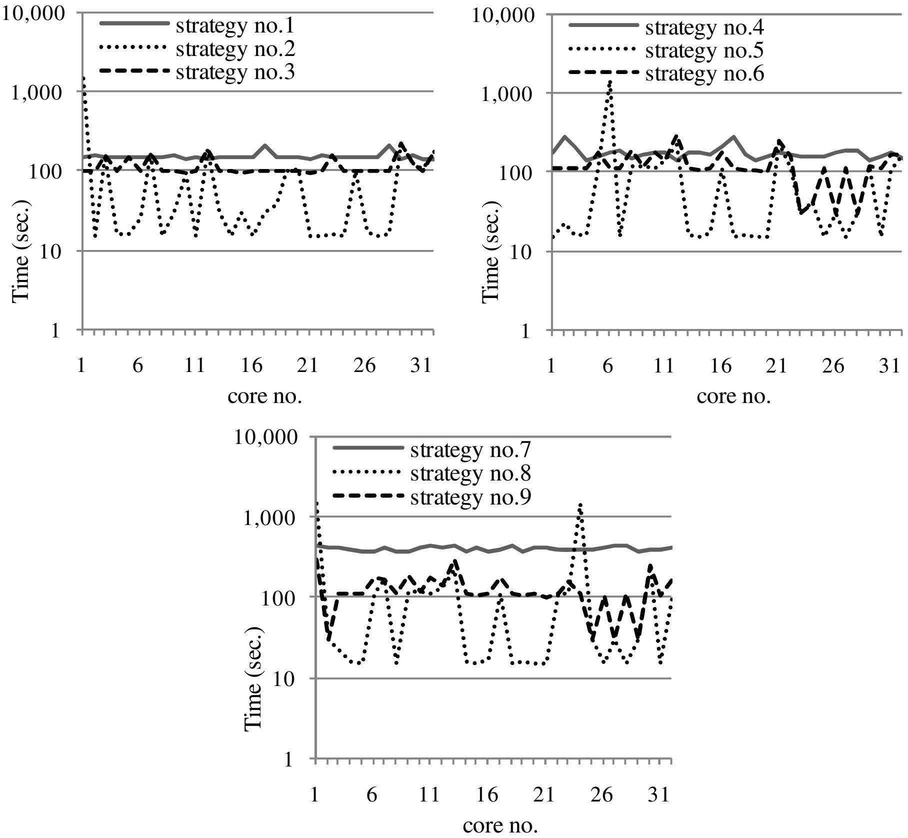


Fig. 5. Workload of parallel FI-growth of the scheduling variation strategies on 32 cores

6 Conclusions

In this paper, we propose that the use of good scheduling strategy can helps speeding up the execution of parallel FI-growth data mining algorithm. We propose a simple step in generating the strategies and then building a tool to evaluate the merit of each strategy on a real task graph obtained from the application on a target multicore cluster system. The results show that various strategies can result in different level of performance on as scalable multicore cluster systems that scale from 8-64 cores.

For the future work, we are working on the improvement of scheduling algorithm. In addition, communication scheduling will be added in order to increase the total system performance. With the right guide-line for parallel data mining algorithm design, we hope to increase the speed and performance of data mining for future data intensive applications.

References

1. Agrawal, R., Imielinski, T., Swami, A.: Mining Association Rules between Sets of Items in Large Databases. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 207–216 (1993)
2. Agrawal, R., Srikant, R.: Fast Algorithms for Mining Association Rules. In: Proceedings of the 20th International Conference on Very Large Data Bases, pp. 487–499 (1994)
3. Agarwal, R.C., Aggarwal, C.C., Prasad, V.V.V.: A tree projection algorithm for generation of frequent item sets. J. Parallel Distrib. Compute. 61, 350–371 (2001)
4. Han, J., Pei, J., Yin, Y.: Mining Frequent Patterns without Candidate Generation. In: Proc. of the ACM SIGMOD International Conference on Management of Data, pp. 1–12 (2000)
5. Amphawan, K., Surarerks, A.: An Approach of Frequent Item Tree for Association Generation. In: Proc. of the IASTED Conference on Artificial Intelligence and Soft Computing (2005)
6. Zaki, M.J.: Parallel and Distributed Association Mining: A Survey. IEEE Concurrency 7(4), 14–25 (1999)
7. Agrawal, R., Shafer, J.C.: Parallel Mining of Association Rules. IEEE Transactions on Knowledge and Data Engineering 8(6), 962–969 (1996)
8. Park, J.S., Chen, M., Yu, P.S.: An Effective Hash-Based Algorithm for Mining Association Rules. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 175–186 (1995)
9. Zaïane, O.R., El-Hajj, M., Lu, P.: Fast Parallel Association Rule Mining without Candidacy Generation. In: Proceedings of the IEEE International Conference on Data Mining, pp. 665–668 (2001)
10. Javed, A., Khokhar, A.: Frequent Pattern Mining on Message Passing Multiprocessor Systems. Distributed and Parallel Databases 16(3), 321–334 (2004)
11. Skillicorn, D.B.: Strategies for parallel data mining. IEEE Concurrency 7, 26–35 (1999)
12. Manaskasemsak, B., Benjamas, N., Rungsawang, A., Surarerks, A., Uthayopas, P.: Parallel association rule mining based on FI-growth algorithm. In: Proceedings of the International Conference Parallel and Distributed System, vol. 2, pp. 1–8 (2007)
13. Srikant, R.: Synthetic Data Generation Code for Association and Sequential patterns. IBM Quest, `http://www.almaden.ibm.com/cs/projects/iis/hdb/Projects/data_mining/datasets/syndata.html`

Bringing Precision to Office Document Search by Semantic Relationship Approach

Somchai Chatvichienchai[1] and Katsumi Tanaka[2]

[1] Dept. of Information and Media Studies, University of Nagasaki, Nagasaki, Japan
somchaic@sun.ac.jp
[2] Dept. of Social Informatics, Kyoto University, Kyoto, Japan
ktanaka@i.kyoto-u.ac.jp

Abstract. Traditional search tools that employ keyword and phrase matching between the query and search index alone tend to offer high recall and low precision. The search users are faced with too many irrelevant results. In order to solve this problem, we propose a novel search technique that effectively searches the target documents by the search query whose definition is based on the document type, search terms and the semantic relationship between the search terms and the target documents. We present a technique that collects search terms and their semantic relationship from office documents and generates XML-based search indices. The search system implementation and query response time evaluation are also discussed.

Keywords: Indices, Documents, Search, Semantic Relationship, XML.

1 Introduction

A major change over the last decade is that the value of unstructured, text-based information has increased substantially as enterprises try to enhance their competitive position through information and knowledge. This kind of information can take the form of text, HTML, meta-data, email, Word documents, spreadsheets etc. Information of enterprises is a critical strategic asset because it is applied by persons of enterprises to complete a task and to manage their organizations. Therefore finding the document files that fit the people's need from disparate data sources is a challenging task. Traditional search tools[2, 7, 18] based on conventional information retrieval techniques tend to offer high recall and low precision. One reason is that these tools do not provide a means that defines the semantic relationship between documents and their search terms. For sake of readability, we use "*semantic relationship of a search term*" to denote "*semantic relationship between a search term and the document containing it*". Consider the search request that finds the purchase orders taken by the salesperson whose name is *Tanaka*. This request cannot be fulfilled by these tools because they cannot identify (*i*) the documents which are purchase order, and (*ii*) the search term "Tanaka" which is regarded as "the name of the salesperson" of the documents. As the result, traditional search of these tools is both under- and over-inclusive, insofar as the query result may miss many critical documents or captures many irrelevant documents.

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 48–57, 2009.

In order to solve the above problem, we propose a novel search system that effectively searches the target documents by the search queries whose definition is based on the document types (such as meeting minutes, sale reports, contracts, letters, etc), search terms and their semantic relationship. We present a technique that collects search terms and their semantic relationship from the documents of some office applications to generate the XML-based search indices that can effectively locate the office documents. The semantic relationship between a document and its search terms is defined by creating a *search term schema* for the document. The proposed technique can be applied to the documents edited by Word documents and Excel spreadsheets of Microsoft Office Suites[12] and PDF documents created by Adobe LiveCycle Designer[1]. In this paper, we use the term "*office documents*" to denote the documents edited by these software programs.

The main contribution of this paper is summarized as follows.

1. We propose a technique that extracts search terms and their semantic relationship from office documents in order to provide an effective search for these office documents.
2. We present schema models that define the basic logical structures of XML schemas for search terms and search indices.
3. We present a method that transforms a given search query into a set of XPath queries on the search index.

The rest of the paper is organized as follows. Section 2 presents the architecture of the proposed search system. Section 3 presents the schema model of our search term schemas and search indices. In section 4, we present the new search query and a technique computing the answer for a given search query. Section 5 describes the evaluation of query response time. In section 6 we discuss related work. Finally, the last section concludes this paper.

2 System Architecture

Figure 1 shows the architecture of the proposed search system. The system consists of four processes: Template Design, Document Creation / Modification, Search Index Generation, and Document Search.

Template Design

For each document type, the system manager creates a template which users use to create a new office document of that type. The system manager defines a search term schema describing search terms of that document type. Due to space limitation, the paper presents a sample method that defines search terms of a purchase order which is edited by Excel. Excel has a Schema Library to which the system manager can add multiple XML schemas[15]. The system manager uses a visual interface of Excel to map elements of a schema which defines search terms of the purchase order to fields of the template. The system manager embeds the VBA macro[9] developed by us to the template.

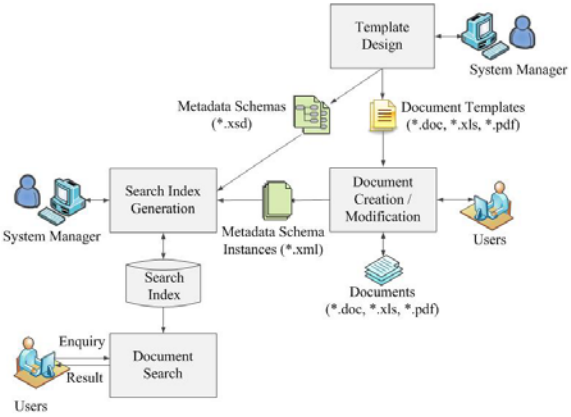


Fig. 1. System architecture

Document Creation / Modification
In this system, users create new office documents from the templates of the previous process. The VBA macro of the document automatically outputs its search terms of the document to an XML[17] file when the document is saved.

Search Index Generation
The search index generation program gathers the outputted XML files of the previous process to create and update search indices of the system. As the search indices are represented in XML format, they can be easily imported to other search engines.

Document Search
The search program requests query users to specify the document type that they want to search. The program looks up the definition of search terms of the specified document type from the search index and automatically generates a query form for that document type. The query form allows users defining a search condition by selecting search terms, and adding operators such as equals, greater than, less than, in the list or between to further expand or restrict the search scope.

3 The Schema Models for Search Term Schemas and Search Indices

In order to provide a guideline for the users to define a schema of search terms (search term schema, for short) of a document type, we propose a *schema model* that defines the basic logical structure of search term schemas. Fig.2(a) shows a schema model of search term schema where *t* denotes the root node whose name specifies a document type. Node g_i ($1 \le i \le n$) denotes a schema element whose name specifies the group name of child nodes $e_{i,1}$, $e_{i,2}$, .. , e_{i,m_n}. The occurrence of node g_i is one or greater than one. Node $e_{i,j}$ ($1 \le i \le n$ and $1 \le j \le m_n$) denotes the schema element of a search term. The value of node $e_{i,j}$ specifies a search term while the name of node $e_{i,j}$

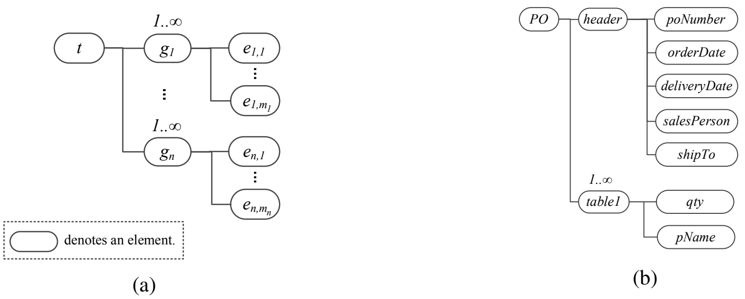


Fig. 2. (a) A model of search term schema and (b) A sample of search term schemas

specifies the semantic relationship between the search term and the document type specified by node *t*. The occurrence of node $e_{i,j}$ is one.

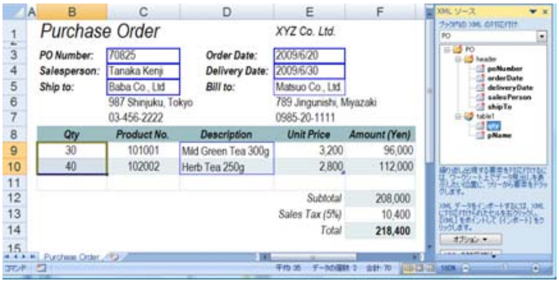
Based on the proposed schema model, we present an example of a search term schema (see Fig.2(b)) that defines the search terms of a purchase order. Figure 3(a) depicts a worksheet to which the schema of Fig.2(b) is bound. Schema node *poNumber* shown in the XML source pane is mapped to the cell which presents a purchase order number. Schema node *qty* is mapped to a list of cells each of which presents the ordered quantity of a product item. We employ the VBA macro of Excel to output search terms of the worksheet into a *Search Term Schema Instance File* (*STIF*, for short). Figure 3(b) shows a sample of *STIF* outputted from the worksheet of Fig.3(a). Based on element *poNumber*, the text string “70825” is regarded as the purchase order number of PO (purchase order) file.

3.1 Search Term Definition File

The search index generation program creates *Search Term Definition File* (*STDF*, for short) and *search indices*. *STDF* provides definition of search terms of the document types gathered by the system. The query program refers *STDF* in order to generate a query form for the document type specified by a query user. Figure 4(a) depicts the schema tree of *STDF*. The search term definition of a document type is automatically inserted into *STDF* when the search index of that document type is first created. The value of attribute *id* of *dType*, *gp* and *fd* denotes a document type, group name and search term name, respectively. Figure 4(b) illustrates an instance of *STDF*. The values of element *fd* and its attribute are set with the element name of *STIF*. The system manager subsequently updates the values of elements *fd* in order to provide more explanation of the search terms.

3.2 Search Indices

The search index generation program constructs a search index for each document type so that query response time of a document type is not affected by the index sizes of other document types. Figure 4(c) shows the schema tree of search indices. Each search index consists of subtrees rooted by *file*. Each subtree contains the URL



	A	B	C	D	E	F
1		Purchase Order			XYZ Co. Ltd.	
3		PO Number:	70825	Order Date:	2009/6/20	
4		Salesperson:	Tanaka Kenji	Delivery Date:	2009/6/30	
5		Ship to:	Baba Co., Ltd	Bill to:	Matsuo Co., Ltd	
6			987 Shinjuku, Tokyo		789 Jingunishi, Miyazaki	
7			03-456-2222		0985-20-1111	
8		Qty	Product No.	Description	Unit Price	Amount (Yen)
9		30	101001	Mild Green Tea 300g	3,200	96,000
10		40	102002	Herb Tea 250g	2,800	112,000
11						
12					Subtotal	208,000
13					Sales Tax (5%)	10,400
14					Total	218,400
15						

繰り返し出現する要素を対応付けるには、ワークシート上でデータ見出しを表示したい位置に、ツリーから要素をドラッグします。

XML データをインポートするには、XML に対応付けられたセルを右クリックし、[XML] をポイントして [インポート] をクリックします。

(a) XML Source Task Pane

```
<?xml version="1.0" encoding="UTF-8"?>
<PO>
    <header>
        <poNumber>70825</poNumber>
        <orderDate>2009-06-20</orderDate>
        <deliveryDate>2009-06-30</deliveryDate>
        <salesPerson>Tanaka Kenji</salesPerson>
        <shipTo>Baba Co., Ltd</shipTo>
    </header>
    <table1>
        <qty>30</qty>
        <pName>Mild Green Tea 300g</pName>
    </table1>
    <table1>
        <qty>40</qty>
        <pName>Herb Tea 250g</pName>
    </table1>
</PO>
```

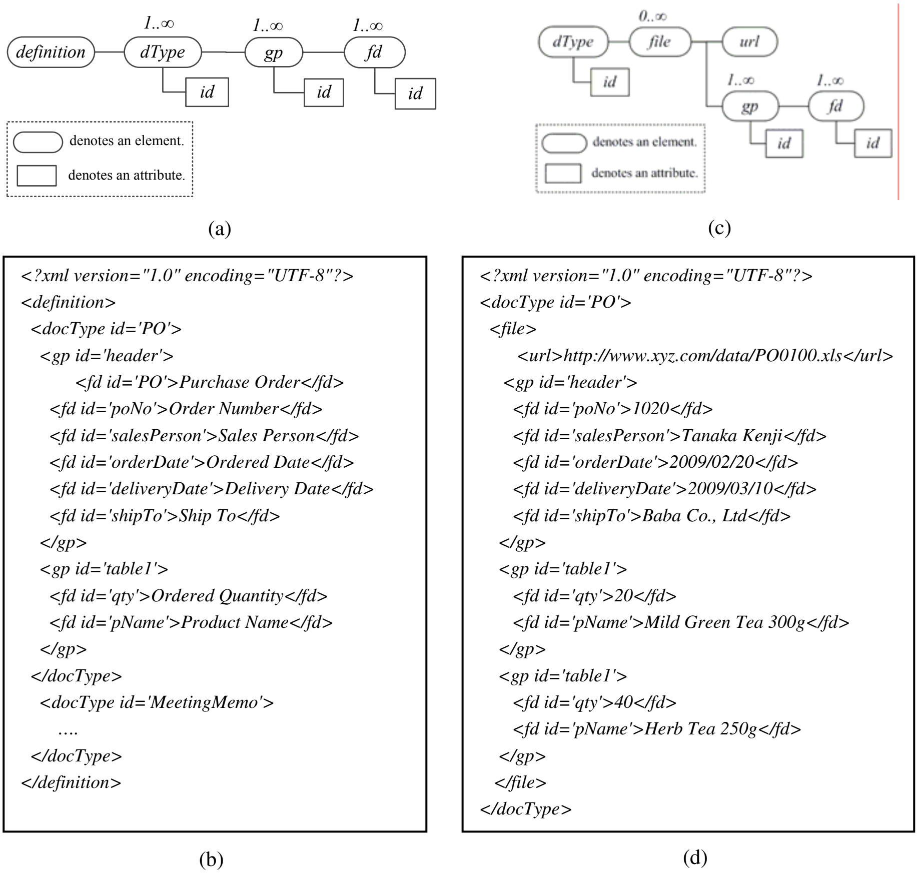
(b)

Fig. 3. (a) Binding the search term schema of Fig.2(b) to a spreadsheet. (b) An XML file presenting search terms of the spreadsheet of (a).

address of an office document and search terms of the documents. Figure 4(d) depicts an example of a search index for purchase order. *<fd id='pName'>Mild Green Tea 300g</fd>* states that the semantic relationship between search term "*Mild Green Tea 300g*" and *PO0100.xls* is *pName* (product name). Note that explanation of *pName* is defines by *<fd id='pName'>Product Name</fd>* of *STDF* of Fig.4(b).

4 Search Query

Search query of this system consists of search conditions whose definition are as follows.



(a)

(c)

```
<?xml version="1.0" encoding="UTF-8"?>
<definition>
 <docType id='PO'>
  <gp id='header'>
      <fd id='PO'>Purchase Order</fd>
   <fd id='poNo'>Order Number</fd>
   <fd id='salesPerson'>Sales Person</fd>
   <fd id='orderDate'>Ordered Date</fd>
   <fd id='deliveryDate'>Delivery Date</fd>
   <fd id='shipTo'>Ship To</fd>
  </gp>
  <gp id='table1'>
   <fd id='qty'>Ordered Quantity</fd>
   <fd id='pName'>Product Name</fd>
  </gp>
 </docType>
  <docType id='MeetingMemo'>
    ....
 </docType>
</definition>
```

(b)

```
<?xml version="1.0" encoding="UTF-8"?>
<docType id='PO'>
 <file>
     <url>http://www.xyz.com/data/PO0100.xls</url>
   <gp id='header'>
    <fd id='poNo'>1020</fd>
    <fd id='salesPerson'>Tanaka Kenji</fd>
    <fd id='orderDate'>2009/02/20</fd>
    <fd id='deliveryDate'>2009/03/10</fd>
    <fd id='shipTo'>Baba Co., Ltd</fd>
  </gp>
  <gp id='table1'>
    <fd id='qty'>20</fd>
    <fd id='pName'>Mild Green Tea 300g</fd>
  </gp>
  <gp id='table1'>
    <fd id='qty'>40</fd>
    <fd id='pName'>Herb Tea 250g</fd>
  </gp>
 </file>
</docType>
```

(d)

Fig. 4. (a) Schema tree of a search term definition file, (b) A sample of search term definition files, (c) Schema tree of a search index and (d) a sample of search indices

Definition 1 (Search Conditions): A *search condition* is defined to have the following syntax.

$$(dtype.gp.fd\ opr\ val),$$

where

- *dtype* denotes a document type;
- *gp* denotes the group name of search terms;
- *fd* denotes a search term which is organized under *gp;*
- *opr* denotes a comparison operator =, >, <, >=, <=, !=, or ⊇; and
- *val* denotes a text string.

We use * to represent a wildcard form for *gp* and *fd*. The comparison operator ⊇ denotes "contains". □

Definition 2 (Search Queries): Based on propositional logic, a *search query* is represented by the following syntax.

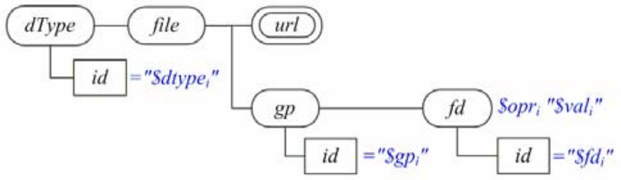


Fig. 5. A tree pattern representing XPath expression for search condition c_i

$$[\neg]C_1\ op_1\ [\neg]C_2\ op_2 \ldots op_{n-1}\ [\neg]C_n,$$

where

- C_i $(1 \le i \le n)$ is a search condition,
- $\neg$ denotes the negation sign,
- $op_i \in \{\wedge, \vee\}$ where $(1 \le i \le n\text{-}1)$ denotes a propositional operator. □

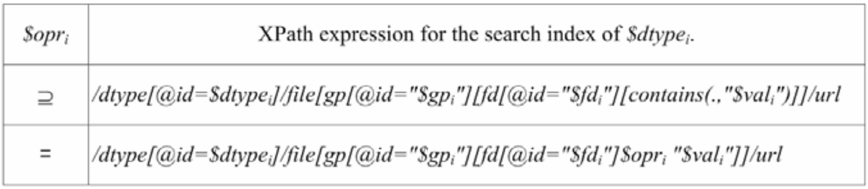
Note that an argument which appears in brackets [ ] is optional.

Example 1: Refer to the search index of Fig.4(d), (*PO.table1.pName* ⊒ "*Green Tea*") ∧ (*PO.table1.qty* > "*30*") denotes a search query that finds *PO* (purchase order) files, whose *pName* (product name) of *table1* group contains the text string "*Green Tea*" and *qty* (ordered quantity) of the same group is greater than 30.

Since each search index is stored as an XML file, XPath[14] expression is employed to locate the URL addresses of the office document files satisfied by a search query. However, the definition of a search condition is different from that of XPath expression. Therefore, the search program needs to transform a search condition of a search query in an XPath expression for the search index. We present a method that translates a search condition into an XPath expression as follows.

Let $C_i \in q$ *be* a search condition of *q*. Let $\$dtype_i$, $\$gp_i$, $\$fd_i$, $\$opr_i$, $\$val_i$ be the document type, group, search term, operator, and value arguments, respectively of C_i $(1 \le i \le n)$. Figure 5 shows a tree pattern of search condition C_i. The node which is enclosed with double line denotes the output node. Based on value of opr_i, the tree pattern of search condition C_i is translated into the XPath expression shown in table 1.

Table 1. Translation of search condition c_i into an XPath expression

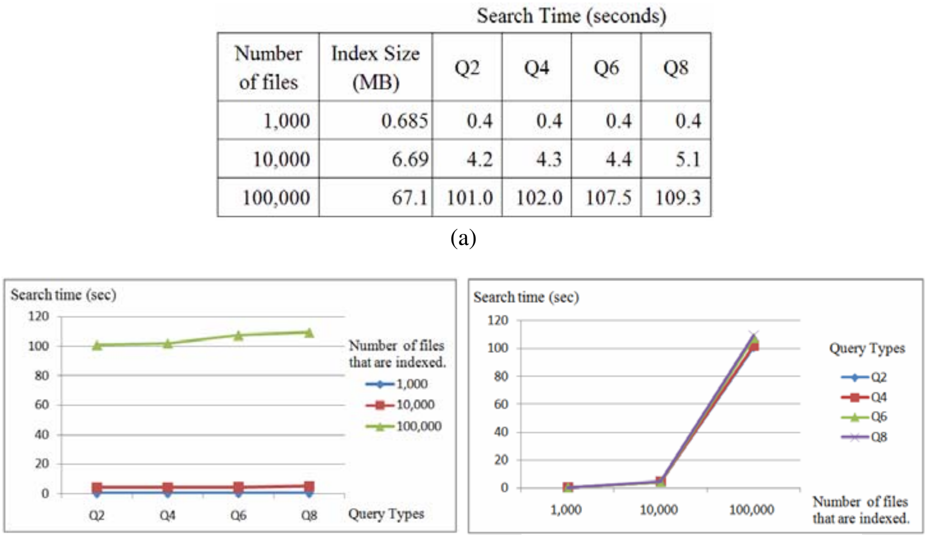


$\$opr_i$	XPath expression for the search index of $\$dtype_i$.
⊒	*/dtype[@id=$\$dtype_i$]/file[gp[@id="$\$gp_i$"][fd[@id="$\$fd_i$"][contains(.,"$\$val_i$")]]/url*
=	*/dtype[@id=$\$dtype_i$]/file[gp[@id="$\$gp_i$"][fd[@id="$\$fd_i$"]$\$opr_i$ "$\$val_i$"]]/url*

5 System Implementation and Evaluation

A common problem of traditional metadata-based search systems is that a search user has to know the metadata needed for searching the desired documents. This problem becomes more seriously in case organizations have many document types and metadata definition varies on the document type. In order to solve this problem, we developed a search program that shows a list of document types which are recorded in the search term definition file. When a user selects a document type that she wants to search from the list, the program looks up metadata definition of the selected document type from the search term definition file and automatically generates query forms of that documents type.

In this system, we store the search indices and the search term definition file in MS Windows Server 2003. Internet Information Services (IIS) 6.0[11] is employed as a Web server which allows users to search documents of their organization via IE web browsers. Our search system is developed by ASP.NET technology[10] in order to efficiently handle dynamically-generated web pages. Figure 6(a) depicts search times of queries of 4 types on three search indexes. Q2, Q4, Q6 and Q8 denote the queries whose numbers of search conditions are two, four, six and eight respectively. The number of the output files satisfied by these queries is one hundred. The experiment is done at a server whose CPU is Pentium® 3.4GHz with main memory 1GB. As shown in Fig.6(b), the number of search conditions gives little impact on the search time of the same search index. However, Fig.6(c) illustrates that the query response time increases drastically with increased size of the search index. Since the search system generates a search index file for searching documents of the same type, size expansion of a search index of a document type has no impact on the search time of other document type.



Search Time (seconds)

Number of files	Index Size (MB)	Q2	Q4	Q6	Q8
1,000	0.685	0.4	0.4	0.4	0.4
10,000	6.69	4.2	4.3	4.4	5.1
100,000	67.1	101.0	102.0	107.5	109.3

(a)

(b) (c)

Fig. 6. Query response time comparison

6 Related Work

Recently there has been a surge of interest in topics of searching local files driven by Google Desktop Search [7] and others [2, 18]. These systems work by creating an index of the files found on a computer (or network) and allowing users to perform keyword searches across the data. These systems, while powerful, do not give the users any input on how their files will be organized and presented. So even though users can find files that contain information on "Profit Report", for example, they do not have a way of describing how the word "Profit Report" is regarded as a data value of the field “Agenda Title” in these files while our work provides a way to describe it.

The work by [8] proposes a technique for automatically generating qualified Dublin Core metadata [5] on a web server. The metadata is structured using the Resource Description Framework (RDF)[16] and expressed in XML. The description covers ten out of fifteen standard metadata. The metadata elements are title, creator, subject, description, publisher, date, format, identifier, relation, and rights. The metadata relation is used to represent a resource that is hyper-linked from the current resource. However, this work focuses on Web documents which are described by HTML. Then this work cannot be applied with unstructured documents.

Semantically enhanced search was addressed from other perspectives, as in Stuff I’ve Seen[6], where contextual terms (e.g., access time, or author) are used to enrich search results, or as in Swoogle[3], in which information retrieval capabilities are offered for semantic documents residing on the Web. The work by [4] shows how search technology can be enhanced with implicit predicates, in order to take into account the structure and semantics defined by applications. The work of [13] proposes a model for knowledge representation of analyzed documents using linguistic concepts and properties. In contrast, our search mechanism provides an effective method that defines index terms for unstructured documents. Therefore, the search index of this paper enables users to pose a search query that meets their intention better than those of conventional search tools.

7 Conclusion and Future Work

We have proposed a novel technique that collects search terms and their semantic relationship from office documents, and generates XML-based search indices. The search indices enable users to effectively find the target office documents thru search conditions defined by document types, search terms and their semantic relationship. We have presented a formal definition of the new search query. The search index generation program constructs a search index for each document type so that query response time of a document type is not affected by the index sizes of other document types. We have also proposed a method that transforms a given search query into a set of XPath queries on the search index.

Our research leaves space for future work. Issues to be investigated include the following two issues. The first is to reduce search time by storing the search index into a native XML database which can provide a more efficient method of computing the output of XPath queries. The second is to develop a method that automatically binds the relevant search term schema to the office documents that have not yet been indexed.

References

1. Adobe Systems Incorporated, Adobe LiveCycle Designer ES (2009), http://www.adobe.com/products/livecycle/designer/
2. Apple, "Spotlight" (2009), http://www.apple.com/macosx/features/300.html, #spotlight
3. Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R.S., Peng, Y., Reddivari, P., Doshi, V.C., Sachs, J.: Swoogle: A search and metadata engine for the semantic web. In: Proc. of CIKM 2004, pp. 652–659 (2004)
4. Dittrich, J.-P., Duda, C., Jarisch, B., Kossmann, D., Vaz, M.A.: Salles ETH Zurich. Bringing Precision to Desktop Search: A Predicate-based Desktop Search Architecture. In: Proc. of ICDE 2007, pp. 1461–1465 (2007)
5. Dublin Core Metadata Initiative, DCMI Metadata Terms (2006), http://dublincore.org/documents/dcmi-terms/
6. Dumais, S., Cutrell, E., Cadiz, J., Jancke, G., Sarin, R., Robbins, D.C.: Stuff i've seen: A system for personal information retrieval and re-use. In: Proc. of SIGIR 2003, pp. 72–79 (2003)
7. Google, Google Desktop Search (2009), http://desktop.google.com/en/GB/features.html
8. Jenkins, C., Inman, D.: Server-Side Automatic Metadata Generation using Qualified Dublin Core and RDF. In: Proc. of Int. Conf. on Digital Libraries, pp. 245–253 (2000)
9. Korol, J.: Excel 2003 VBA programming with XML and ASP. Wordware Publishing (2006)
10. Liberty, J., Hurwitz, D.: Programming Asp. Net, Oreilly & Associates Inc. (2003)
11. Microsoft, Technical Overview of Internet Information Services (IIS) 6.0 (2002), http://www.microsoft.com/windowsserver2003/techinfo/overview/iis.mspx
12. Microsoft, Microsoft Office Suites (2009), http://office.microsoft.com/en-us/suites/HA101757031033.aspx
13. Rinaldi, A.M.: A content-based approach for document representation and retrieval. In: DocEng 2008, pp. 106–109. ACM Press, New York (2008)
14. W3C, XML Path Language (XPath) Version 1.0 (1999), http://www.w3.org/TR/1999/REC-xpath-19991116
15. W3C, XML Schema (2001), http://www.w3c.org/XML/Schema
16. W3C, Resource Description Framework, RDF (2004), http://www.w3.org/RDF/
17. W3C, Extensible Markup Language (XML) 1.0 (4th edn.) (2006), http://www.w3.org/TR/2006/REC-xml-20060816/
18. X1 Technologies, X1 Professional Client (2009), http://www.x1.com/products/xds.html

Multi-objective Optimization for Information Sharing in Vehicular Ad Hoc Networks

Grégoire Danoy[1], Bernabé Dorronsoro[1], Pascal Bouvry[1], Bojan Reljic[1],
and Frank Zimmer[2]

[1] Faculty of Science, Technology & Communication, University of Luxembourg
firstname.lastname@uni.lu
[2] SES-Astra TechCom, Luxembourg
frank.zimmer@ses-astra.com

Abstract. Satellites to car multimedia content delivery systems, such as KU Mobile, are a promising area for research and development. With vehicles becoming more computational and communicative, the demand for bringing multimedia and information technology services into vehicles is raising. However, these systems suffer from significant signal fading and incomplete reception caused by obstacles along the road interrupting the mandatory line of sight between satellite and car while these two are communicating. Therefore, additional technologies need to be specifically designed to complete partially received information in such scenarios. In this paper we propose xChangeMobile, an inter-vehicular ad-hoc wireless network (VANET) content exchange system. It is composed by two novel communication protocols, VanetDFCN and ChunkXChange, designed specifically for this scenario. Finally, using multi-objective genetic algorithms, we optimized the main parameters of proposed protocols to achieve the best communication performances.

Keywords: Vehicular Ad Hoc Networks, Multi-Objective Optimization, Cellular Genetic Algorithm.

1 Introduction

Vehicular ad hoc networks (VANETs) are self-organized communication networks spontaneously arising between communication capable vehicles without any previously existing infrastructure. One of the main interests of this kind of networks is the possibility of coupling them with other kinds of fixed networks that can provide additional services and information that could help our driving to be safer, more efficient, and more pleasant. This includes warning applications, e.g. cars able to send warning messages to other cars alerting them of a danger ahead, weather or traffic conditions, etc.

There currently exist some initiatives for extending the functionality of VANETs by using fixed networks. As an example, the CALM (Continuous Air-interface for Long and Medium range telecommunications initiative) [1] set of ISO standards aims to define a standardized set of air interface protocols and

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 58–70, 2009.

parameters for medium and long range communications. The main objective is to provide continuous communications between a vehicle and roadside stations using several different communication media, such as cellular, microwave, millimeter, or infrared. Another example is the car-to-car communication consortium [2], which aims to standardize wireless communications interfaces and protocols between vehicles and their environment in order to make vehicles of different manufacturers interoperable and also enable them to communicate with road-side units.

There also exist in the literature several approaches using satellites in order to provide new services to VANETs. An example is the KU Mobile project [3,4], a satellite-to-vehicle multimedia content delivery system, implemented in a common project by experts from satellite, vehicles, and telecommunication companies. The system developed in the KU Mobile project focuses primarily on the streaming of multimedia files for near-real-time use: in the reception devices, each packet of a service packet is stored until all packets corresponding to one file are correctly received.

The main problem with the use of satellite communications for VANETs is that no obstacles can be in the communication line going from the satellite to the receivers, and this restriction is not always met in the case of VANETs, since roads have usually many possible obstacles for the satellite communication, such as bridges, tunnels, buildings, trees, etc. In the frame of the KU Mobile project, the lost packages due to communication failures can be retrieved through redundant packets, which are re-transmitted within short time delays. This way, signal interruptions leading to packet losses do not immediately degrade the service quality. However, introducing redundancy does not solve the problem in all cases, unfortunately, since the number of repetitions of the same information is limited and small in number, and in case of long *shadowing* (i.e. non-coverage) periods, like in urban areas or large tunnels, certain parts of content could fail to be received in all repetitions (and thus the multimedia file cannot be played since there is some *gap* in cached packets), or at least it would delay too much the reception of the file, lowering the QoS of the system.

In order to deal with this loss of packets due to obstacles, some proposals exist in the literature. They usually rely on Complementary Ground Components (CGCs), i.e., on the location of terrestrial receivers in order to broadcast the satellite streaming in low coverage regions, such as cities [5]. However, this approach could be valid in large non-covered and highly populated regions (like cities), but it would be very difficult to cover any non-covered area with CGCs. The contribution of this paper is to design a protocol for VANETs in order to overcome the gaps in the multimedia recieved files without using any existing infrastructure, which would be a much more realistic approach than the use of CGCs. Our protocol was designed as an extension of the existing KU Mobile system, although it can be generalized to any streaming protocol for satellites. The proposed protocol is called xChangeMobile, and it is designed to allow content exchange based on wireless ad hoc communication in order to fill the gaps in the devices cache memories.

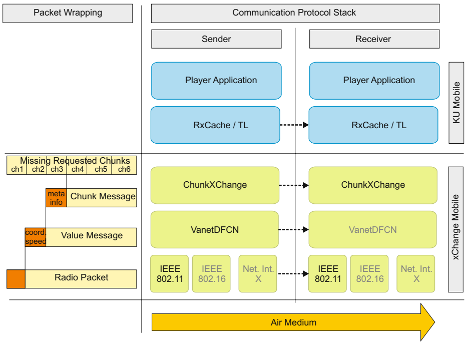


Fig. 1. xChangeMobile Communication Protocol Stack and messages types

The details of xChangeMobile as well as the related optimization problem will be presented in the following section. Section 3 provides a detailed description of the multiobjective metaheuristic used, i.e. MOCell. In section 4 the optimization of the protocol's behavior is presented and simulation results are discussed. Finally, we end in Section 5 with our main conclusions and further research lines.

2 xChangeMobile

The xChangeMobile protocol was designed for the exchange of the missing parts between devices in a VANET to which some source is streaming large multimedia files (e.g., TV, radio, etc.). These missing parts are due to some occasionally coverage loss between devices and the source that could be produced by vehicles mobility and/or the presence of obstacles blocking the communication. We assume in this work that KU Mobile transmits multimedia files using TCP packets. These TCP packets are grouped in chunks by xChangeMobile, so a chunk is just a container, in which TCP packets are stored. It is composed of two parts: a ChunkID which is a unique identifier (ID) placed in the header of the Chunk and a ChunkContent which is composed of one up to many TCP packets. The size of chunks is a fixed value of our protocol, and it must be the same for all devices (more details are provided in Section 2.2).

XChangeMobile is designed to work on three layers: PHYsical - communication technology (for example: IEEE 802.11, see Fig. 1), MAC (VanetDFCN) and APPlication layer. Every layer abstracts a communication with matching layer of the other communication party, and besides vertical communication interface, it is not aware of any other detail of underlying/overlying layer. Every layer has its own format of messages, which encapsulates the message of the above layer. Finally, the highest layer of xChangeMobile, the ChunkXChange protocol, communicates with the lowest layer of the streaming protocol. In the case of the KU Mobile system we consider in this paper, the lowest layers interacting with XChange Mobile are RxCache and Transport Layer (RxCache/TL) —as it is displayed in Fig. 1.

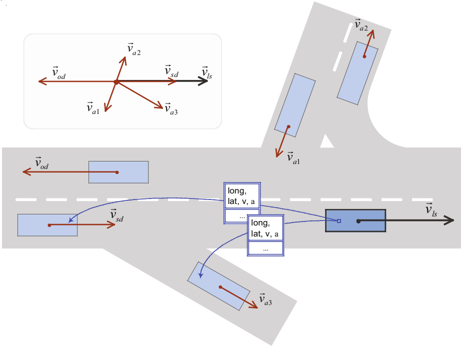
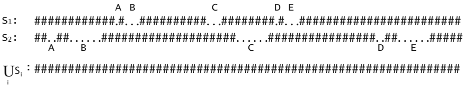


Fig. 2. VanetDFCN takes into account vehicle's location, speed, and direction (information embedded in packets' headers) to estimate for how long they will be in range

The idea of xChangeMobile is based on the assumption that two neighbor stations have the same, or very similar, pattern of gaps (missing chunks of information) in local caches, although usually with different time offset and scale size. The effect is that the missing parts in one local cache would be available in some of the neighboring devices in ideal case, and then they could communicate for exchanging the missing content. However, if some of these parts are still missing in the neighbor devices, these neighbor devices will be waiting for them from other devices, and they will forward the information after receiving it in case it is still needed.

We illustrate in Fig. 3 a typical situation that occurs when two vehicles equipped with KU Mobile receivers drive one after the other. We can imagine a vehicle travelling fast equipped by receiver S1, which is in a given time in range with another vehicle (with S2 receiver) that travels at a lower speed (e.g., S1 is overtaking S2). In this case, the time offset and difference in scaling effects, are clearly visible: gaps, denoted as A, B, C, D and E, exist in both receiver cache memories, where S2 cache is affected probably by the same obstacles than S1, but earlier in time, and for a longer time period, since S2 is driving ahead and slower than S1.

The two following sections introduce two novel communication protocols, namely VanetDFCN and ChunkXChange, that we defined for xChangeMobile. As it can be seen in Fig. 1, the VanetDFCN protocol is in a lower level of abstraction than ChunkXChange. Thus, ChunkXchange is in charge of managing the information, requesting and sending chunks, and updating the cache memory, while VanetDFCN is a communication protocol used for the information exchange among cars.



```
                   A  B                    C            D  E
S1:   ##############.#...############...##########.#...#############################
S2:   ##..##......########################......####################..##......######
         A      B                             C                           D      E

⋃Si : ##############################################################################
 i
```

Fig. 3. Example of the cache memory of two devices S1 and S2 moving at different speeds. Symbols "#" and "." stand for cached and missing messages, respectively.

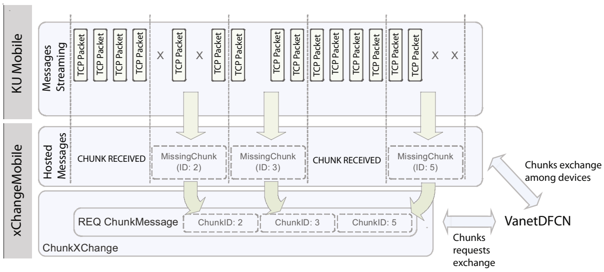


Fig. 4. Management of missing chunks performed by ChunkXChange

2.1 VanetDFCN

VanetDFCN is a broadcasting protocol designed for inter-vehicular communication in Vehicular Ad hoc Networks (VANETs). It is an extension of DFCN (Delayed Flooding with Cumulative Neighborhood) [6], a broadcasting protocol designed for metropolitan Mobile Ad hoc Networks (MANETs). DFCN is extended in VanetDFCN by adding new criteria for deciding whether it is worth to forward a received message or not. The objective is to reach as many stations as possible while optimizing the network use.

VanetDFCN introduces global geographical location awareness to communication. It assumes the presence of a global positioning system device (GPS) that is used to obtain location (longitude, latitude) and movement related information (velocity, direction). This information is added to the header of every message that the station transmits using the VanetDFCN protocol.

The functioning of VanetDFCN is shown in Algorithm 1. We consider to have two messages queues: in incomingMessageQueue we can find all the received messages, while outgoingMessageQueue contains the list of messages to be sent by the next lower layer protocol. At each iteration of VanetDFCN, the protocol chooses the messages to be processed among all the received ones (those in

Algorithm 1. Iteration of VanetDFCN at every node

```
Data: incomingMessageQueue, outgoingMessageQueue
for msg IN incomingMessageQueue do
    if (msg.DTL <getDistanceHopped(msg)) AND isWorthyCommunicating() AND
    firstTimeReceived(msg) then
        remove(incomingMessageQueue,msg);
        hostedMessages <– msg;
        push_up(msg); // Send it to ChunkXChange (higher level protocol)
    else
        drop(msg);
    end if
end for
for msg IN hostedMessages do
    outgoingMessageQueue <– DFCN. sendMessage(msg);
end for
```

incomingMessageQueue); the rest of messages will be discarded. Each message must meet the following three conditions to be further processed: first, the maximum distance to live (DTL), which is a parameter of the protocol indicating the maximum distance the message can travel, cannot be exceeded; second, the message must be received for the first time; and third, the protocol must decide whether it is worth to communicate or not with the receiver of the message. For taking this last decision, an estimation of the time the two devices (sender and receiver) are in range is made in terms of their position (latitude and altitude), movement speed and direction (see Fig. 2). Then this estimation is used to compute the number of chunks they could exchange, and if this number of chunks is higher than a given threshold value (fixed by the protocol), then the communication is considered to be worthy. Once a message meets the previous three conditions, it is removed from the incomingMessageQueue, it is added to the hostedMessages list, and then it is sent to the ChunkXChange protocol. In other case, if at least one of the three conditions is not met, then the message is discarded. After processing all the messages in the incomingMessageQueue, then the protocol will ask DFCN to forward all the messages in hostedMessages.

2.2 ChunkXChange

The ChunkXChange protocol is designed to deal with the chunks management. Specifically, the three main functions of this protocol are (i) making requests for chunks that are missing in the local cache memory, (ii) answering to possible chunk requests received from other stations, and (iii) updating the cache memory when an answer to a request is received. If the received chunk completes a given file, then this file is ready to be played, so it is copied to the local memory and removed from the cache. Thus, the objective of this protocol is to fill the gaps of missing chunks by requesting the information to other devices.

The behavior of the ChunkXChange protocol is shown in Fig. 4. As it was explained in Section 2, TCP messages received from KU Mobile are merged together into chunks, the size of chunks being specified by the protocol. Every received chunk (both from KU Mobile and VanetDFCN) is stored in the cache memory. Then, in every request for missing chunks (chunks with one or more TCP packets lost), ChunkXchange includes the IDs of all the missing chunks in the local cache memory at that moment (the request is sent through VanetDFCN). Then, when a given station receives a request of a given number of chunks, it answers by sending the available requested chunks in its local memory through VanetDFCN.

2.3 Optimization Problem

The xChangeMobile protocol is based on several parameters that markedly influence its behavior. Hence, in our simulation process we decided to indentify these parameters and find their most suitable values in order to optimize the behavior of the proposed protocol. These parameters are shown in Table 1. Among them, only the broadcast message lifetime was not described in previous sections. It

Table 1. Configurable parameters of the xChangeMobile protocol

Parameter	Allowed Values	Units	Protocol
Broadcast Message Lifetime	2-100	Seconds	VanetDFCN
DTL	0-1000	Meters	ChunkXChange
ReRequest Period	5-60	Seconds	ChunkXChange
ChunkMessage Size	1-100	Chunks	ChunkXChange
Density Threshold	1-1000	Stations	VanetDFCN (DFCN)

represents the time (expressed in seconds) for how long a message is valid for further processing (storing in the cache, transmitting over the network, etc.). Once the lifetime of a message expires, the message is immediately discarded. This parameter is used to deal with the limited size of the local cache of every device on the VanetDFCN level.

For measuring the quality of our protocol with the different parameterizations we set two different objectives to be minimized. They are the number of remaining missing chunks in all the vehicles after the simulation process and the bandwidth usage, measured as the total number of messages sent in the network. Thus, we need to rely in multi-objective optimization [7,8] for solving this problem with two conflicting objectives. We say they are in conflict because optimizing one of them means decreasing the quality of the other one.

3 Multiobjective Cellular Genetic Algorithms

This section introduces the multiobjective metaheuristic used for solving the introduced problem. Section 3.1 introduces the basics of the canonical cellular genetic algorithm and section 3.2 provides a decription of MOCell, a multiobjective algorithm based on a cGA model.

3.1 Canonical cGA

Cellular GAs [9,10,11] are structured population algorithms with a high explorative capacity. The individuals composing their population are arranged into a (usually) two dimensional toroidal mesh, and only neighbor individuals (i.e., the closest ones measured in Manhattan distance) are allowed to interact during the breeding loop (see Fig. 5). This way, we are introducing some kind of isolation in the population that depends on the distance between individuals. Hence, the genetic information of a given individual can be spread slowly through the grid (since neighborhoods are overlapped), and it will need a high number of generations to reach distant individuals (thus preventing the population from premature convergence). Structuring the population this way, we achieve a good exploration/exploitation tradeoff on the search space, thus improving the capacity of the algorithm for solving complex problems [12].

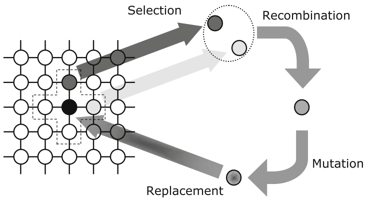


Fig. 5. In cellular GAs, individuals are only allowed to interact with their neighbors during the breeding loop

A canonical cGA follows the pseudo-code included in Algorithm 2. In this basic cGA, the population is usually structured in a regular grid of d dimensions ($d = 1, 2, 3$), and a neighborhood is defined on it. The algorithm iteratively applies the variation operators to each individual in the grid (line 3). An individual may only interact with individuals belonging to its neighborhood (line 4), so its parents are chosen among its neighbors (line 5) with a given criterion. Crossover and mutation operators are applied to the individuals in lines 6 and 7, with probabilities P_c and P_m, respectively. Afterwards, the algorithm computes the fitness value of the new offspring individual (or individuals) (line 8), and inserts it (or one of them) into the place of the current individual in the population (line 9) following a given replacement policy. This loop is repeated until a termination condition is met (line 2). The most usual termination conditions are to reach the optimal value (if known), to perform a maximum number of fitness function evaluations, or a combination of them.

We graphically show in Figure 5 the different steps performed during the breeding loop in cGAs for every individual, already explained above. There are two possible ways for updating the individuals in the population [13]. The one shown in Figure 5 is called asynchronous, since the newly generated individual is inserted back into the population (following a given replacement policy) immediately after its creation, and thus it can interact in the breeding loop of its neighbors, even when they most probably belong to previous generations.

Algorithm 2. Pseudocode for a Canonical cGA

```
 1: proc Steps_Up(cga)      //Algorithm parameters in 'cga'
 2: while not Termination_Condition() do
 3:    for individual ← 1 to cga.popSize do
 4:       n_list←Get_Neighborhood(cga,position(individual));
 5:       parents←Selection(n_list);
 6:       offspring←Recombination(cga.Pc,parents);
 7:       offspring←Mutation(cga.Pm,offspring);
 8:       Evaluate_Fitness(offspring);
 9:       Insert(position(individual),offspring,cga);
10:    end for
11: end while
12: end proc Steps_Up;
```

Algorithm 3. Pseudocode of MOCell

```
proc Steps_Up(mocell)     //Algorithm parameters in 'mocell'
Pareto_front = Create_Front() //Creates an empty Pareto front
while !TerminationCondition() do
   for individual ← 1 to mocell.popSize do
      n_list←Get_Neighborhood(mocell,position (individual));
      parents←Selection(n_list);
      offspring←Recombination(mocell.Pc,parents);
      offspring←Mutation(mocell.Pm,offspring);
      Evaluate_Fitness(offspring);
      Replace(position(individual),offspring,mocell, aux_pop);
      Insert_Pareto_Front(offspring, individual);
   end for
   mocell.pop←aux_pop;
   mocell.pop←Feedback(mocell,ParetoFront);
end while
end_proc Steps_Up;
```

However, there is also the possibility of updating the population in a synchronous way, meaning that all the individuals in the population are updated at the same time. For that, an auxiliary population with the newly generated individuals is progressively built in every generation, and after applying the breeding loop to all the individuals, the current population is replaced by the auxiliary one [9].

3.2 Multiobjective cGA

In this section we describe the MOCell metaheuristic, a multiobjective algorithm based on a cGA model. As described in [14], it follows the canonical synchronous cGA (see Algorithm 2). The pseudocode of the algorithm is given in Algorithm 3. It can be observed that Algorithms 2 and 3 are very similar. One of the main differences between the two algorithms is the existence of a *Pareto front* in the multiobjective case.

Indeed, in multi-objective optimization we cannot generally speak about better or worse solutions since we are dealing with several objectives at the same time (one solution can be worse than another one for some objectives but better for some other ones). Thus, the concept of *dominating* solutions is introduced: one solution dominates another one if the former is better than the latter for all the considered objectives. In other case we call them non-dominated solutions, since we cannot state that one solution is better than the other. Therefore, the result of a given multiobjective problem is not a single best solution but a set of non-dominated ones, which is called the *Pareto front*.

In MOCell, the Pareto front is just an additional population (the external archive) composed of a number of the non-dominated solutions found during the search process. The archive has a maximum size, so a density estimator is need to remove solutions when it becomes full; MOCell uses the crowding distance, proposed for NSGA-II [15], for that purpose.

MOCell starts by creating an empty Pareto front (line 2 in Algorithm 2). Individuals are arranged in a 2-dimensional toroidal grid, and the genetic operators are successively applied to them (lines 7 and 8) until the termination condition is met (line 3). Hence, for each individual, the algorithm consists of selecting two parents from its neighborhood, recombining them in order to obtain an offspring, mutating it, and evaluating the resulting individual; then the algorithm decides whether the new offspring replaces the current one (line 10). The next step (line 11) is to insert the offspring into the external archive, if appropriate. Finally, after each generation, the old population is replaced by the auxiliary one, and a feedback procedure is invoked to replace a fixed number of randomly chosen individuals of the population by solutions from the archive. As in the case of a single-objective cGA, four asynchronous versions MOCell can be obtained by changing the way the cells are updated.

In this algorithm, the resulting offspring replaces the individual at the current position if the latter is better than the former, but, as it is usual in multiobjective optimization, we need to define the concept of "best individual". Our approach is to replace the current individual if it is dominated by the offspring or both are non-dominated and the current individual has the worst crowding distance (as defined in NSGA-II) in a population composed of the neighborhood plus the offspring. This criterion is also used to decide if the offspring solutions are added to the external archive (line 11 in Algorithm 3). For inserting individuals in the Pareto front, the solutions in the archive are also ordered according to the crowding distance; then, when inserting a non-dominated solution, if the Pareto front is already full, the solution with a worst crowding distance value is removed.

MOCell has been implemented in Java using the jMetal framework [14]. It can be obtained in the Web from: `http://jmetal.sourceforge.net`.

4 Simulation and Optimization of the Protocol

For evaluating the fitness value of individuals, we needed to rely on simulations. Specifically, we used the Madhoc [16] simulator for measuring the quality of the solutions. Additionally, we needed DFCN, which was already implemented in Madhoc, which was indeed successfully optimized using Madhoc in previous works [17,18,9]. This simulator had to be modified in order to include the KU Mobile and the xChangeMobile protocols. For testing the behavior of our system, we have defined a highway scenario in the Madhoc simulator. Specifically, we set 10 vehicles moving at speeds between 60 km/h and 150 km/h in one road of 2.5 km length, and 20% of the road surface is out of coverage from satellite. We consider devices to have a range in the interval [450, 500] meters [19], and the communication bandwidth is 5.5 mbps [19]. The simulation runs for 60 seconds.

As previously mentioned in section 3, for this optimization process we used the MOCell algorithm, implemented in the jMetal framework [20,21], that has been proven to outperform the current state-of-the-art algorithms such as NSGA-II and SPEA2 [14,22,9]. Due to the hard computational requirements of the fitness

Broadcast Message Lifetime (sec.)	Messages to Transmit Threshold (msgs.)	Distance to Live (m.)	Re-Request Period (sec.)	Chunk Message Size (chunks)	Band Ussage (msgs.)	Still Missing Chunks (%)
2.3	**998.2**	**8.2**	**59.8**	**84.5**	**59.4**	**86.5**
2.3	992.4	3.5	60.0	82.7	63.7	82.6
2.3	993.3	6.0	59.9	76.9	66.0	80.8
2.0	998.8	7.7	59.8	70.0	79.7	73.7
2.3	994.2	53.5	59.3	76.5	86.9	66.1
2.0	990.7	17.8	59.9	65.2	90.2	58.6
2.3	960.6	53.9	60.0	58.5	92.5	55.5
2.3	991.2	6.0	59.9	51.7	98.7	50.1
2.3	**999.3**	**15.4**	**60.0**	**50.5**	**99.0**	**50.1**
2.3	998.1	15.4	60.0	49.3	100.1	49.7
2.3	872.0	7.8	59.9	54.6	103.6	45.5
2.3	885.2	21.6	59.9	51.7	108.4	42.6
2.3	498.7	22.4	59.9	54.2	119.2	39.5
4.8	551.8	34.1	59.1	57.4	132.6	37.0
4.9	454.6	29.4	59.6	46.8	139.4	33.3
8.0	500.1	35.7	59.0	56.2	150.6	31.4
11.5	565.4	22.4	59.9	54.1	166.3	29.0
11.1	**418.3**	**10.1**	**59.9**	**49.8**	**167.5**	**29.0**

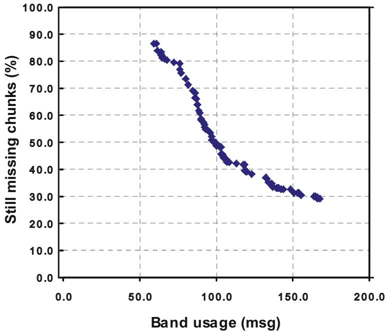


Fig. 6. Pareto front with non-dominated solutions

function (we need to run Madhoc to simulate the network behavior), we set the final condition of our algorithm to 5000 evaluations.

The results are shown in Fig. 6, where the plot represents the best non-dominated solutions found, and the table summarizes a selection of the solutions found. As expected, the two objectives we are optimizing are in conflict, and choosing a solution with better band usage means decreasing the number of still missing chunks. Hence, we propose here two different solutions (in grey background in the table in Fig. 6) in terms of the importance we give to the two objectives. The last row of the table should be chosen if we give more importance to the percentage of still missing chunks, since less than 30% of chunks in all the devices are lost. The second solution is proposed in the case we would like to have a good compromise between the two objectives. This solution has some parameter values similar to those in the first row (e.g., the broadcast message lifetime, and the messages to transmit threshold), while the chunk message size is similar to the solution in the last row, and the distance to live for messages has an intermediate value between those of the extreme solutions. As it can be seen, one interesting result is that for every solution the re-request period is always around 60 seconds. This means that it is an appropriate value independently of the importance we give to the objectives.

Analyzing the obtained solutions, we can see that around half of them have more than 50% of still missing chunks in vehicles' cache memories. We believe that this undesirable result is a consequence of the short simulation time we were forced to set in this preliminary approach (60 seconds is really too short time for sending long multimedia files). The reason for this short simulation time we set is the hard computational requirements of the problem, since it took more than 15 hours to solve it, and increasing the simulation time leads to a considerably increment of the time to evaluate solutions. In further research works, we consider that it is mandatory to increase the simulation time for obtaining better results.

In order to get solutions in reasonable time we are working on parallelizing the algorithm in large clusters of computers or even in grids.

5 Conclusions and Future Work

We proposed in this paper a new protocol for exchanging missing file chunks between devices in vehicular ad hoc networks. The scenario considered is a satellite streaming multimedia files to cars, and these cars exchange among them the missing chunks to overcome possible coverage failures during the reception due to tunnels, trees, buildings, etc. The protocol was implemented in Madhoc, an ad hoc network simulator, and some key parameters of the protocol were identified. The values assigned to these parameters could highly influence the behavior of our protocol. Hence, a multi-objective algorithm was used to find the best assignments to these parameters in order to optimize the behavior of the protocol in terms of the network usage and the number of still missing chunks in the vehicles after the simulations. Three different parameter assignments were proposed in this work depending on the importance we give to the two optimization objectives.

As future works, we plan to deep more in this optimization process, evaluating other different multi-objective algorithms, and increasing in the simulations the number of devices and the surface. Furthermore, we are also considering the design of new protocols for the same purpose, these protocols could be based on clustering methods or on spanning trees, for instance.

Acknowledgment

F. Zimmer and B. Reljic acknowledge that this work has been partially founded by SES-Astra.

References

1. CALM, Continuous communication for vehicles: http://www.calm.hu/ (last accessed in May 2008)
2. Car to car communication consortium: http://www.car-to-car.org/ (last accessed in May 2008)
3. The KU Mobile project: http://telecom.esa.int/telecom/www/object/index.cfm?fobjectid=13103 (last accessed in May 2008)
4. KU Mobile specification documentation, Ses-astra internal documentation (2008)
5. Evans, B., Thomson, P.: Aspects of satellite delivered mobile TV (SDMB). In: 16th IST Mobile and Wireless Communications Summit, pp. 1–5. IEEE Press, Los Alamitos (2007)
6. Hogie, L., Seredynski, M., Guinand, F., Bouvry, P.: A bandwidth-efficient broadcasting protocol for mobile multi-hop ad hoc networks. In: International Conference on Networking (ICN), p. 71. IEEE Press, Los Alamitos (2006)

7. Coello, C., Van Veldhuizen, D., Lamont, G.: Evolutionary Algorithms for Solving Multi-Objective Problems. In: Genetic Algorithms and Evolutionary Computation. Kluwer Academic Publishers, Dordrecht (2002)
8. Deb, K.: Multi-Objective Optimization using Evolutionary Algorithms. John Wiley & Sons, Chichester (2001)
9. Alba, E., Dorronsoro, B.: Cellular Genetic Algorithms. In: Operations Research/Compuer Science Interfaces. Springer, Heidelberg (2008)
10. Whitley, D.: Cellular genetic algorithms. In: Forrest, S. (ed.) Fifth International Conference on Genetic Algorithms (ICGA), p. 658. Morgan Kaufmann, California (1993)
11. Manderick, B., Spiessens, P.: Fine-grained parallel genetic algorithm. In: Schaffer, J. (ed.) Third International Conference on Genetic Algorithms, pp. 428–433. Morgan Kaufmann, San Francisco (1989)
12. Alba, E., Tomassini, M.: Parallelism and evolutionary algorithms. IEEE Transactions on Evolutionary Computation 6(5), 443–462 (2002)
13. Alba, E., Dorronsoro, B., Giacobini, M., Tomassini, M.: Decentralized Cellular Evolutionary Algorithms. In: Handbook of Bioinspired Algorithms and Applications, ch. 7, pp. 103–120. CRC Press, Boca Raton (2006)
14. Nebro, A.J., Durillo, J.J., Luna, F., Dorronsoro, B., Alba, E.: Mocell: A cellular genetic algorithm for multiobjective optimization. International Journal of Intelligent Systems 24(7), 726–746 (2009)
15. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elist Multiobjective Genetic Algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6(2), 182–197 (2002)
16. Hogie, L., Guinand, F., Bouvry, P.: The Madhoc Metropolitan Adhoc Network Simulator. Université du Luxembourg and Université du Havre, France (2006), http://www-lih.univ-lehavre.fr/~hogie/madhoc/
17. Alba, E., Dorronsoro, B., Luna, F., Nebro, A., Bouvry, P., Hogie, L.: A cellular multi-objective genetic algorithm for optimal broadcasting strategy in metropolitan MANETs. Computer Communications 30(4), 685–697 (2007)
18. Luna, F., Nebro, A., Dorronsoro, B., Alba, E., Bouvry, P., Hogie, L.: Optimal broadcasting in metropolitan MANETs using multiobjective scatter search. In: Rothlauf, F., Branke, J., Cagnoni, S., Costa, E., Cotta, C., Drechsler, R., Lutton, E., Machado, P., Moore, J.H., Romero, J., Smith, G.D., Squillero, G., Takagi, H. (eds.) EvoWorkshops 2006. LNCS, vol. 3907, pp. 255–266. Springer, Heidelberg (2006)
19. Günter, Y., Grobmann, H.: Usage of wireless LAN for inter-vehicle communication. In: Proc. of the 8th IEEE International Intelligent Transportation Systems, pp. 408–413. IEEE, Los Alamitos (2005)
20. Durillo, J., Nebro, A., Luna, F., Dorronsoro, B., Alba, E.: jMetal: A java framework for developing multiobjective optimization metaheuristics. Technical Report ITI-2006-10, Dpto. de Lenguajes y CC.CC., Universidad de Málaga (2006)
21. Durillo, J.: : The jMetal framework, http://neo.lcc.uma.es/software/metal (last accessed in May 2008)
22. Nebro, A., Durillo, J., Luna, F., Dorronsoro, B., Alba, E.: A study of strategies for neigborhood replacement and archive feedback in a multiobjective cellular genetic algorithm. In: Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., Murata, T. (eds.) EMO 2007. LNCS, vol. 4403, pp. 126–140. Springer, Heidelberg (2007)

Analysis on Relationships among Software Models through Traceability Activity

Waraporn Jirapanthong

Faculty of Information Technology
Dhurakij Pundit University
110/1 -4 Prachachuen Road,
Laksi, Bangkok 10210, Thailand
waraporn@it.dpu.ac.th

Abstract. In this paper, we present the architecture of traceability activity to assist the analysis on the relationships between the software models, particularly the ones being generated during the analysis and design phases of software product family system development. We describe the software models being created during the analysis and design phases. We also present the components of traceability activity on those software models. Moreover, we discuss the extension of XQuery as the rule language for representing traceability rules based on our experiences in traceability activity.

Keywords: Software Traceability, Requirements Traceability, XQuery.

1 Introduction

The current trend in software systems is product families rather than single products. Nowadays customers appeal to products that have a sense of uniqueness, products that are compatible but slightly different than those of their friends. The answer from industry is to set-up flexible product lines, which include a range of disciplines from product development to product manufacturing. The efficiency of these product lines for evolving systems is mainly determined by the amount and ease of reuse of existing artefacts.

During the analysis and design phase of a product, it has always been significant and will increasingly be so. The reasons are (i) the growth of the complexity of systems, and (ii) the trend towards product families. However, the techniques and approaches to supporting the software specification are not well-defined or standardized. According to [3], our work concentrates on software models generated during the phases of analysis and design. Particularly, the approach includes two main essentials: (i). the types of documents represented software models created during the phase of domain analysis; and (ii) the types of documents represented software models created during the phase of domain design. We apply a feature-based approach which is important to support domain analysis and domain design, enhance communication between customers and developers in terms of product features, and assist with the development of software product line architecture. On the other hand, an object-oriented approach is necessary to assist with the development of the various product

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 71–80, 2009.

members. As the following section, we elaborate the idea of applying featured-based objected-oriented engineering approach.

In this paper, we address the difficulty of analysis on the relationships between the software models. In order to be practically applicable in industry, it is required that such a framework suits the development organisation, builds on proven technology, and that its application is non-intrusive. We present the enabling of traceability activity to assist the analysis on the relationships between the software models. This leads the benefits such as (i) enhancing the functionality of a product or adapting it to a changed environment, (ii) checking a consistent set of development models, and (iii) providing a starting point for a more structured approach to reuse. Particularly, we present the framework of software models being generated during the design and analysis phase of product family systems. Moreover, we present an approach which is based on the practice of traceability activity to allow automatic generation of traceability relations between the software models.

Software traceability is the ability to relate software artefacts created during the life-cycle of software system development such as retrieval documents, requirements specifications, analysis and design models, source codes, and test cases [1, 2, 4, 5, 6]. However, traceability practice becomes more difficult and ambiguous in product family systems due to their rigidness and complexity.

The remaining of this paper is structured as follows. In section 2, we present the software model specification for product family systems. In section 3 we describe the traceability on software models presented in section 2 by presenting the classification and generation of traceability relations as well as discussing the extension of a rule language for traceability activity. Finally, in section 4 we conclude and discuss the work.

2 Software Model Specification for Product Family Systems

Our work concentrates on the specification of artefacts generated during the phases of domain analysis and domain design. Particularly, the approach includes two main essentials: (i). the types of documents represented software models created during the phase of analysis; and (ii) the types of documents represented software models created during the phase of design. Additionally, we believe that a feature-based object-oriented engineering approach is required when developing product line systems. A feature-based approach is important to supporting domain analysis and design, enhances communication between customers and developers in terms of product features, and assists with the development of software product line architecture. On the other hand, an object-oriented approach is necessary to assisting with the development of the various product members. As the following section, we elaborate the idea of applying featured-based objected-oriented engineering approach. We describe each type of software models. We also give examples by referring the system domain of mobile phone systems.

2.1 Requirements Artefacts

The requirements artefacts created during the analysis phase is represented by feature model and use case. In the following, we described the details.

Use_Case *UseCaseID*="**UC1**" *System*="MobilePhone" *Product_Member*="**PM1**"
Existential *Commonality_Variability*="Alternative"
  Variant_Point v1
  Variants v1 {keying-in a phone number of a receiver, selecting a phone number from a list of contacts}
Title *Sending a Message*
Description The phone is able to send a text message. The user can specify an address of a receiver by selecting from a list of contacts.
Level User Goal
Preconditions The user has already selected function of sending a text message from the main menu.
Postconditions The phone has sent the message.
Primary_actor The user
Secondary_actors -
Flow_of_events
  Event 1 The system shows an editor for writing a message.
  Event 2 The user inputs a phone number by [v1].
  Event 3 The system displays the phone number to which the message is being sent.
  Event 4 The user enters the message and confirms sending the message.
  Event 5 The system sends the message and displays an acknowledge on the screen.
Exceptional_events -
Superordinate_use_case –
Subordinate_use_case –

Fig. 1. An example of a use case

Use case. In our work, we express the requirements of product line systems by extending the use case definition given by Cockburn (Cockburn 2000). A use case is composed of: (1) **Use_Case** – the element consists of three attributes, which are information of the use case: (a) *Use_Case_ID*, (b) *System* – this attribute specifies which domain of product line is, and (c) *Product_Member* – this attribute specifies for which product member the use case is specified; (2) **Existential** – this element is used to represent the existential of a use case. It consists of an attribute *Commonality_Variability* – this attribute can be *mandatory*, *alternative*, and *optional*. The attribute *Commonality_Variability* is specified as "alternative", the element **Existential** can consist of sub-element **Variant_Point** which specifies a particular point of the use case's variability. The element can consist of a sub-element either **Variant** or **Parameter**. The element *Variant* specifies a set of alternatives for the particular variant point, as the element *Parameter* specifies the domain of the *Variant_Point*; (3) **Title** – the element *Title* is the title of use case; (4) **Description** – the element *Description* is specified for a brief textual description; (5) **Level** – the element describes the level of functionality that it describes within a system; (6) **Preconditions** – the element describes the conditions that must be satisfied before its execution; (7) **Postconditions** – the elements describes the conditions that must be satisfied after its execution; (8) **Primary_actors** – the element specifies primary users of the use case; (9) **Secondary_actors** – the element specifies secondary users of the use case; (10) **Flow_of_events –** the element specifies a list of the events that trigger the use case and the specification of the normal events that occur within it. The element *Flow_of_events* consists of the sub-element **Event**, which specifies a particular event being preceded in the use case; (11) **Exceptional events** – the element describes the events that do not always occur when the use case is executed; (12) **Superordinate use case** – the element specifies a use case for which the use case is elaborated; and (13) **Subordinate use cases** – the element specifies a use case to which the use case is specified.

Figure 1 illustrates an example of a use case *Sending a Message* from a mobile phone for product member PM1 of the mobile phone case study.

Feature Model. We proposed to apply the feature model presented in FORM (Kang et al. 1998) which is based on the feature model proposed by (Kang et al. 1990). The authors enhanced the feature model with a textual specification for each feature. Our feature model describes the requirements artefacts of a product line system and illustrates the features available in the line.

A feature is represented by a name and can be (i) *mandatory*, when it must exist in the applications in the domain; (ii) *optional*, when it is not necessary to be presented in the applications in the domain; or (iii) *alternative*, when it can be selected for an application from a set of features that are related to the same parent feature in the hierarchy.

The features can be classified into four groups namely (i) *application capabilities*, signifying features that represent functional aspects of the applications; (ii) *operating environments*, signifying features that represent attributes of the environment in which product members are used and operated; (iii) *domain technologies*, signifying features that represent specific implementation and technological aspects of the applications in the domain; and (iv) *implementation techniques*, signifying features that represent more general implementation and technological aspects of the applications, but not necessary specific for the domain.

Feature can also be related by different types of relationships. Examples of these relationships are (i) *composed_of*, (ii) *generalisation/specialization*, and (iii) *implemented_by* relationship types. Figure 2 presents an example of a textual specification for feature *Text Messages*.

Feature-name:	Text Messages
Description:	The phone can edit, send, and receive a short text message
Issues and decision:	Text message over mobile phone is a way of communication
Type:	Application capability
Commonality:	Mandatory
Composed-of:	Sending Text Messages, Receiving Text Messages, Editing Text Messages
Composition-rule:	-
Allocated-to-subsystem:	Messaging

Fig. 2. An example of a textual specification for feature *Text Messages*

2.2 Design Artefacts

In our approach, we adopt UML class diagram, statechart diagram, and sequence diagram to present the software product line architecture. In the following, we described the details.

Class Diagram. We extend the class diagram presented in (Clauss 2001) by adding some elements. The diagram consists of elements as described following: (1) **Class Diagram** – the element consists of three attributes, which are information of the class diagram: (a) *Class_Diagram_ID,* (b) *System,* and (c) *Product_Member*; (2) **Existential** – this element is used to represent the existential of a class diagram. It consists of an attribute *Commonality_Variability* – this attribute can be (i) *mandatory*, (ii) *alternative*, and (iii) *optional*; (3) **Class** – the element *Class* specifies a system component that is composed of *attributes*, which describe properties of a particular class, and methods, which specify operations of the particular class. A class can be one of three types for expressing variability in product line: (i) *variationPoint*, which represents a variation point of product line, (ii) *variant*, which represents an alternative of a particular variation point, and (iii) *optional*, which represents an optional class; (4) **Relationship** – classes can be associated by applying one of two relationship types: (i) *generalization/specialization*, which associates between classes typed of variationPoint and variant, and (ii) *association with cardinality 0...1*, which associates between any class and a class typed of optional.

State Chart Diagram. The diagram consists of elements as described following: (1) **State Chart Diagram** – the element consists of three attributes, which are information of the state chart diagram: (a) *State_Chart_Diagram_ID*, (b) *System*, and (c) *Product_Member*; (2) **Existential** – this element is used to represent the existential of a state chart diagram. It consists of an attribute *Commonality_Variability* – this attribute can be (i) *mandatory*, (ii) *alternative*, and (iii) *optional*; (3) **State** – the element *State* specifies the system's particular status. We define three types of a state for expressing variability in a product line: (i) *variationPoint*, which represents a state that initiates a variation point of product line, (ii) *variant*, which represents an alternative states of a particular variation point, and (iii) *optional*, which represents an optional state; (4) **Transition** – the element *Transition* describes a driving method to transform a state to another state. To capture and represent variability of a product member, a transition can be specified as one of three transition types: (i) *variantTransition*, which describes one of possible driving methods to transform a state to another state, (ii) *parameterTransition*, which describes a transition requiring a parameter to drive the method, and (iii) *optionalTransition*, which describes a possible driving method to transform a state to another state.

Sequence Diagram. The diagram consists of elements as described following: (1) **Sequence Diagram** – the element consists of three attributes, which are information of the class diagram: (a) *Sequence_Diagram_ID*, (b) *System*, and (c) *Product_Member*; (2) **Existential** – this element consists of an attribute *Commonality_Variability* – this attribute can be (i) *mandatory*, (ii) *alternative*, and (iii) *optional*; (3) **Sequence** –three types of sequences for expressing variability in a sequence diagram: (i) *variationPoint*, which represents a sequence that initiate a variation point of later sequences, (ii) *variant*, which represents an alternative sequence of a particular variation point, and (iii) *optional*, which represents an optional sequence; (4) **Message** – the element *Message* basically represent a called operation from an object interacting to another object. A message can be representing the variability of a product member. Specifically, we propose three types of messages for expressing the variability: (i) *variantMessage*, which is one of possible messages being sent from a *varaintPointSequence* to another sequence, (ii) *parameterMessage*, which is a message requiring a parameter to drive the method, and (iii) *optionalMessage*, which is an optional message that may or may not be sent on a sequence.

3 Traceability on Software Models

We have identified nine different types of traceability relations between the various documents described in the previous section. One or many types of traceability relations can exist between two particular models. A description of each relation is given the following section.

3.1 Classification of Traceability Relations

We describe below these traceability relations types and the types of software models to which the traceability relations exist.

Satisfiability: In this type of relation an element e1 satisfies an element e2, if e1 meets the expectation and needs of e2. For example, a *satisfiability* relation may be hold between an operation or attribute of a class in a class diagram and the description of a use case or the description of a feature in a feature model.

Dependency: In this type of relation an element e1 *depends on* an element e2, if the existence of e1 *relies on* the existence of e2, or if changes in e2 have to be reflected in e1. A *dependency* relation may be hold between the description of a use case and the description of a feature in a feature model.

Overlap: In this type of relation an element e1 *overlaps* with an element e2, if e1 and e2 refer to common aspects of a system or its domain. This is a bi-directional relation. An *overlap* relation may exist between the description of a feature in a feature model and a class in a class diagram, a state in a statechart diagram, or an object or message in a sequence diagram.

Evolution: In this type of relation an element e1 *evolves to* an element e2, if e1 has been replaced by e2 during the development, maintenance, or evolution of the system. An *evolution* relation occurs between document models of the same type for the product member(s) in a family. This relation may hold between elements in use cases, class diagrams, statechart diagrams, and sequence diagrams.

Implements: In this type of relation an element e1 *implements* an element e2, if e1 *executes* or *allows* for the achievement of e2. An *implements* relation may be hold between a sequence of events in a sequence diagram and a feature in a feature model, flow of events in a use case, or the description of a use case.

Refinement: This type of relation associates elements in different levels of abstractions. A refinement relation identifies how complex elements can be broken down into components and subsystems, and how elements can be specified in more details by other elements. Thus, an element e1 *refines* an element e2, when e1 specifies more details about e2. A *refinement* relation may be hold between a message in a sequence diagram and an operation of a class in a class diagram.

Containment: In this type of relation an element e1 *contains* an element e2, when e1 is a document, or an element in a document, that uses an element e2, or a set of elements from a different document. This relation may be hold between sequence or statechart diagrams and classes in a class diagram).

Similar: This type of relation occurs between documents of the same type for different product members. This relation assists with the identification of common aspects between various product members. A *similar* relation is a bi-directional relation that may hold between elements in use cases, class diagrams, statechart diagrams, and sequence diagrams. A *similar* relation between elements e1 and e2 depends on the existence of a relation between e1 and another element e3 and a relation between e2 and element e3. For example, a use case uc1 is *similar* to a use case uc2, if both uc1 and uc2 hold a *containment* relation with a feature f1. Similar relations between two elements can be derived from other relations based on the following inference rules: *Overlap relations*; *Containment relations*; *Satisfiability relations*; *Refinement relations*; *Dependency relations*; and *Implements relations*.

***Different*:** This type of relation also occurs between documents of the same type for different product members. This relation assists with the identification of variable aspects between various product members. More specifically, a different traceability relation expresses the different specialization of a particular variation point between two product members. A *different* relation is a bi-directional relation that may hold between elements in use cases, class diagrams, statechart diagrams, and sequence diagrams. A *different* relation between an element e1 and e2 depends on the existence of a relation between e1 and another element e3, and a relation between e2 and another element e4, where e3 and e4 are variants of the same variability point (e.g. subclasses of the same superclass, sibling features of the same parent feature). For example, a use case uc1 is *different* from a use case uc2, when there are two subclasses c1 and c2 of the same parent class c, where c1 *implements* uc1 and c2 *implements* uc2. Different relations between two elements can be derived from other relations based on the following inference rules: *Overlap relations*; *Containment relation*; *Satisfiability relations*; and *Implements relations*.

3.2 Generating Traceability Relations

Our work proposed the architecture of traceability activity. The architecture is implemented by using a tool which is composed of seven components: (a) to provide the user interfaces for a user to interact with the tool; (b) to identify a set of relevant documents to be traced based on the input from the interface component; (c) to identify a set of traceability rule templates that are related to the documents and relations to be traced based on inputs to the interface component; (d) to verify a traceability rule; (e) to create a set of instantiated traceability rules by replacing the placeholders of the document types in the identified traceability rule templates which the names of the documents to be traced; (f) to generate traceability relations by executing the traceability rules; and (g) to record and present the traceability relations.

Figure 3 shows an example of *satisfiability* relations. One is created since a synonym of *Sending* and *Message* appear in the description of the feature at an appropriate distance. Similarly, another relation is created since a synonym of *Transmitting* and *Message* appear in the description of the feature at an appropriate distance. Thus, the results of rule R86 for use case *UC1* and feature *Text Messages*, and use case *UC2* and feature *Text Messages* are captured and recorded in the relation document.

```
<Relation_Document>
  <Relation RuleID="R86" Type="satisfiability" DocType1="Use Case"
            DocType2="Feature Model">
    <Element Document="file:///c:/UseCase_UC1.xml"> <Title> Sending a Message</Title> </Element>
    <Element Document="file:///c:/Feature_MP.xml">
            <Feature_name> Text  Messages  <Description>…</Description> </Feature_name>
    </Element>
  </Relation>
  <Relation RuleID="R86" Type="satisfiability" DocType1="Use Case" DocType2="Feature Model">
    <Element Document="file:///c:/UseCase_UC2.xml">
        <Title> Transmitting Messages </Title> </Element>
    <Element Document="file:///c:/Feature_MP.xml">
        <Feature_name> Text  Messages <Description> </Description> </Feature_name> </Element>
  </Relation> …
</Relation_Document>
```

Fig. 3. XML-based traceability relations

3.3 Extension of a Rule Language for Traceability Activity

Our approach proposes the extension of XQuery [9] as a rule representation language for traceability rules. Apart from the embedded functions offered by XQuery, it is possible to add new functions and commands. More specifically, we have extended XQuery: (a) to support the representation of parts of the rules, i.e. the actions to be taken when the conditions are satisfied, and (b) to support extra functions to cover some of the traceability relations. These functions have been implemented in XQuery and Java, and are concerned with the identification of specific elements in the documents and words that are synonyms, or textual comparison. We found the consequences of using XQuery for supporting traceability activity. We describe below the advantages and disadvantages of applying XQuery language.

Pros

Powerful to navigate XML documents: XQuery is basically designed to support navigating two different XML documents in the same time. This allows us to determine relationships between two elements in two different documents. Moreover, XQuery is specifically designed for querying XML data which is ordered, nested, hierarchical. This allows us to traverse the hierarchy of XML documents. Those models represent different levels of requirements in the product family systems. The rule, thus, requires traversing XML data in various hierarchies of XML documents.

Support generation of traceability relations in XML format: XQuery includes expressions to construct new XML data which appear as XML fragments. This enables our approach to automatically generate traceability relations and represent them in XML-format.

Handle large size of XML documents and high volume of traceability relations: XQuery is designed to handle manipulation and iteration over sequences of XML data in large-sized XML documents and manage creation the high volume of XML data. An example from our experiments, a rule (e.g. R68) is used to identify *satisfiability* traceability relations between sequence diagram and use case. The rule created 132 relations between sequence diagrams specified in an XMI document (1363 KB) and 8 use cases.

Cons

No support for full text search: XQuery language provides functions and operators for retrieving data in the level of XML elements. However, XQuery does not support full text search. According to our traceability approach, the generation of traceability relations requires comparing the texts of documents. Thus, it can be overloaded on coding in the tool.

No support for synonym, grammatical role, and distance checking: The fact is that the multiplicity of stakeholders participating in the development of the system. This may lead to the use of different words to represent the same thing. Traceability is required to take consideration into the use of equivalent words to specify documents. Traceability is also required to take consideration into the grammatical roles of words in the textual parts of the documents and distance of words being compared in the text. However, XQuery built-in functions and operators do not support those requirements. Extra functions are created to achieve the requirements.

Standardisation is subject to change: The specification of XQuery language is still subject to change and the implementation is incomplete. Currently, our traceability rules are based on XQuery 1.0.

Difficult to create, read, and use: Creation and using of traceability rules in XQuery assume that users are familiar with the concepts of XML technologies (e.g. XML, XPath). XQuery statement that is not easy to understand for unfamiliar users to XML and XQuery. Moreover, composing a XQuery statement is not XML-based. It also requires users familiar with XPath and SQL techniques.

No support for creation and maintenance XML documents: Although XQuery language includes expression to construct XML data, it does not provide commands for creation and updating XML documents (e.g. insert, update, delete). According to our traceability approach, traceability relations are created when the condition part of a rule is satisfied. The traceability relations, thus, are restored into XML document by the tool.

As shown in Table 1, we summarise the requirements of the rule representation language for traceability activity and compare XQuery and other existing techniques for rule representation language i.e. RuleML[7] and XPath [8].

Table 1. Requirements of rule language for traceability activity

Requirements	RuleML	XPath	XQuery
Support input in XML format	√	√	√
Support output in XML format	√	√	√
Support retrieval of data in different documents	√	√	√
Support joining data in different documents	-	√	√
Provide synonym checking	-	-	-
Create XML fragments	√	-	√
Provide aggregated functions to group data	-	-	√
Support update and maintenance XML documents (e.g. insert, update, delete)	-	-	-
Support other XML-based technologies i.e. RDF, XML Schema, XPath, and XLink	√	√	√
Response fast	√	√	√

4 Discussion and Conclusion

In this paper, we presented the architecture of traceability activity that is proposed to perform an automatic generation of traceability relations. We described each type of software models generated during the analysis and design phases of software product system development. We also presented the classification and generation of traceability relations. We analysed using XQuery for traceability activity based on experiences with case studies.

As discussed in previous section, it is difficult to create a traceability rule in XQuery in a case that the semantic of the rule is complicated. The generation of traceability relations need to take consideration thoroughly into documents. More specifically, text in XML documents is parsed as single words in order to verify the conditions of traceability rules. However, XQuery built-in functions and operations do not cover all requirements of traceability activity. Our approach needed to tackle the problems by extending extra functions to satisfy the requirements. As XQuery is designed for retrieving XML data, the language itself is not XML-based. Users require the knowledge of other techniques such as XML, XPath, and SQL. Moreover, the standard of the language is not stable. This can be more difficult for unfamiliar users.

However, regarding to our experience of using XQuery for traceability activity, we found that we can benefit from extending functions in XQuery and other programming language (e.g. Java) to satisfy complicated conditions of traceability rules. Particularly, missed functions and operations can be paid off by extended functions. XQuery supports traversing complicated structure of XML documents and processing a large-sized of documents. It handles creating a large volume of XML data and joining a number of XML data between different XML documents. These characteristics support the activity of traceability that involves a high volume and large size of documents.

References

1. Antoniol, G., Canfora, G., Casazza, G., Lucia, A.D., Merlo, E.: Recovering Traceability Links between Code and Documentation. IEEE Transactions on Software Engineering 28, 970–983 (2002)
2. Gotel, O., Finkelstein, A.: Contribution Structure. In: The Second IEEE International Symposium on Requirements Engineering (RE 1995), pp. 100–107. IEEE Computer Society Press, New York (1995)
3. Jirapanthong, W.: An Approach to Software Artefact Specification for Supporting Product Line Systems, ISBN 978-974-671-574-1, Dhurakij Pundit University, Thailand (2008)
4. Lindvall, M., Sandahl, K.: Practical Implications of Traceability. Software Practice and Experience 26, 1161–1180 (1996)
5. Pohl, K.: Process-Centered Requirements Engineering. John Wiley & Sons, Chichester (1996)
6. Ramesh, B., Jarke, M.: Towards Reference Models for Requirements Traceability. IEEE Transactions on Software Engineering 27, 58–93 (2001)
7. RuleML, `http://ruleml.org/`
8. XPath, `http://www.w3.org/TR/xpath`
9. XQuery, `http://www.w3.org/TR/xquery/`

An Axiomatic and Object-Based Approach to Tracing Safety Properties in the Context of ARP 4754

Paul Attasara-Mason

SIU International University, 16th Floor Shinawatra Tower III, Viphawadi-Rangsit Road,
Chatchuchak, Bangkok, Thailand, 10900
paul@shinawatra.ac.th

Abstract. The complexity of modern systems engineering projects often results in long development cycles, yielding a vast array of artifacts. To maintain stakeholder communication and comply with standards, means to record and navigate links between these artifacts are required. The provision of such mechanisms is termed traceability.

Traceability objectives include, support for: i) management of safety arguments; ii) evolution of artifacts; and iii) impact analysis/change control. Our approach comprises a set of information structures (represented as base axioms in an Object-Oriented and Deductive Database), together with procedures for their analysis. The approach is illustrated using a case study of an aircraft wheel braking system.

Keywords: Traceability, Safety Analysis, Systems Engineering.

1 Introduction

A safety-critical system (SCS) is one whose failure may cause injury or loss of life. Computers are increasingly being used (rather than electromechanical or other components) to control safety-critical applications due to their processing power, size, weight and flexibility - all of which can lead to cost project savings. In this paper we propose an approach for managing safety properties during development and assessment of SCS.

The scale and complexity of modern systems engineering projects, especially those involving SCS, often results in long development cycles. Civil aircraft are a prime example. On October 25 2007, an Airbus A380, the world's first double-deck, widebody, airliner touched down in Sydney following its inaugural commercial flight. This landmark in aviation history came however more than a decade after the A380 project's inception.

Such protracted lifecycles yield a vast array of artifacts, from wiring schematics to piping and instrumentation diagrams, FMECA tables and Fault Trees, reflecting the involvement of engineers from heterogeneous disciplines (e.g. electrical, mechanical and hydraulic, as well as software engineering). To maintain communication among stakeholders [1], manage change [2] and comply with international standards [3, 4], means are required to record and navigate links between these artifacts; such means are known as *traceability*.

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 81–95, 2009.

Our previous output on traceability has centered around MAST (*M*eta-modelling *A*pproach to *S*ystem *T*raceability), a framework which supports the engineering of heterogeneous systems. Work on MAST has been ongoing for the past three to four years, yielding a number of international publications, mostly describing its constituent CASE tools [5, 6, 7]. This paper marks a shift in direction, focusing *less* on tools and *more* on the traceability process. In doing so we are especially concerned with tracing safety properties (attributes of a system which are intended to prevent failure conditions causing hazards). The highly integrated and complex nature of systems under discussion here affords greater opportunity for the introduction of developmental errors which in turn, can lead to undesirable or unintended behaviour which impacts on safety. This scenario is evident in our featured example based on an aircraft braking system.

A means of making the target system and its development/assessment process more visible would increase the detection and elimination of errors. For *development*, this would involve explicitly recording safety requirements imposed over the system and establishing links between these and the system elements that discharge them. For *assessment*, the rationale (arguments) supporting safety claims need to be recorded. Through traceability, this can be provided by structures to record development and assessment products, and their analysis procedures.

We propose an *A*xiomatic approach to *S*ystem *Tr*aceability and *A*nalysis (*ASTrA*) which enables the recording of established categories of safety related information such as hazards and safety arguments, etc. The information is organised in a collection of structures (represented as base axioms in an Object-Oriented and Deductive Database, OODDB), with each structure relating to a particular view, including: system decomposition, safety analysis, decision rationale and the organisation of safety arguments. Analysis is via deductive axioms that verify the definition of populated structures and maintain consistency across projections of this information, together with inference rules for navigation. To illustrate the approach, we demonstrate how results of a Fault Tree Analysis can be encoded and analysed in an information structure and related to structures from other development/assessment activities.

The rest of this paper is therefore organised as follows: section 2 outlines the key traceability concerns for complex SCS; section 3 presents the theory underlying our approach, while section 4 contains a demonstration of its application to a Wheel Braking System; section 5 presents some concluding remarks.

2 Key Traceability Concerns for Safety-Critical Systems

The main component of any traceability approach is a set of structures defining basic information elements and relationships. Two main activities are involved in their manipulation (figure 1): i) *population* - recording development and assessment information; ii) *analysis* - examining their integrity and extracting particular views.

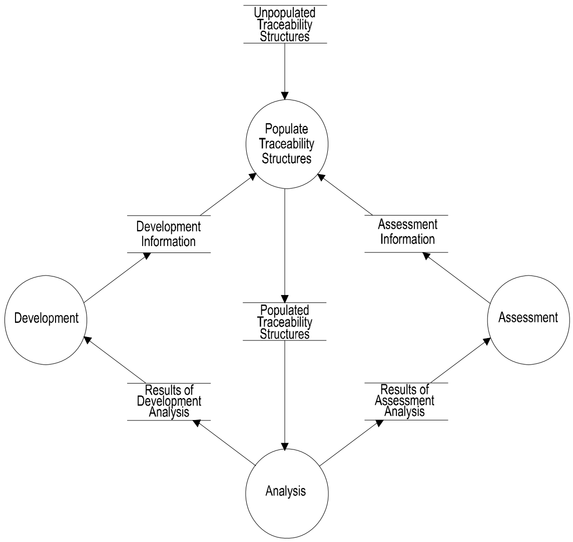

Unpopulated Traceability Structures
Populate Traceability Structures
Development Information
Assessment Information
Development
Populated Traceability Structures
Assessment
Results of Development Analysis
Results of Assessment Analysis
Analysis

Fig. 1. Top Level Trace Process

The populated structures can support a range of traceability concerns (objectives). From the literature and through discussion with practitioners, we have identified several such concerns for SCS stakeholders. These are as follows:-

2.1 Management of Safety Properties

System safety depends partly upon the regime in place to preserve its organisation [8]; i.e., the development and assessment process, the elicitation and documentation process, together with mechanisms supporting traceability of both procedures and artifacts. ARP 4754 prescribes a safety assessment process for ‘Highly-Integrated or Complex Aircraft Systems’ which runs parallel to development. Its sub-processes and respective traceability concerns are as follows: Functional Hazard Assessment (FHA) is conducted at the outset to identify details on possible hazards (e.g., rate and severity) and accidents. The resulting hazard log must therefore support traceability between such entities. To ensure that the reproduction of your illustrations is of a reasonable quality, we advise against the use of shading. The contrast should be as pronounced as possible.

A *Preliminary System Safety Assessment* (PSSA) subsequently defines safety constraints and safety strategies which must be traceable to the hazards they exclude. Finally, the *System Safety Assessment* (SSA) aims to provide evidence for a safety case. Thus, support is required for expressing safety arguments, for tracing them to their respective design components and enforcing consistency and completeness checks between various analysis techniques, e.g. Fault-Tree Analysis and Event-Tree Analysis. *(N.B. A role for traceability within ARP 4754 is shown in figure 2.)*

A *Preliminary System Safety Assessment* (PSSA) subsequently defines safety constraints and safety strategies which must be traceable to the hazards they exclude. Finally, the *System Safety Assessment* (SSA) aims to provide evidence for a safety

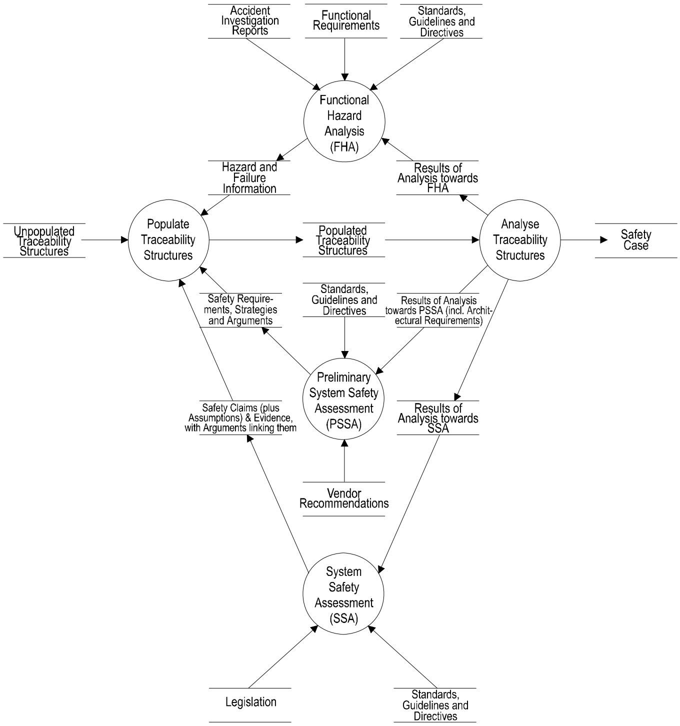

Accident Investigation Reports
Functional Requirements
Standards, Guidelines and Directives
Functional Hazard Analysis (FHA)
Hazard and Failure Information
Results of Analysis towards FHA
Unpopulated Traceability Structures
Populate Traceability Structures
Populated Traceability Structures
Analyse Traceability Structures
Safety Case
Safety Require-ments, Strategies and Arguments
Standards, Guidelines and Directives
Results of Analysis towards PSSA (incl. Archit-ectural Requirements)
Preliminary System Safety Assessment (PSSA)
Safety Claims (plus Assumptions) & Evidence, with Arguments linking them
Results of Analysis towards SSA
Vendor Recommendations
System Safety Assessment (SSA)
Legislation
Standards, Guidelines and Directives

Fig. 2. Safety Assessment Trace Process

case. Thus, support is required for expressing safety arguments, for tracing them to their respective design components and enforcing consistency and completeness checks between various analysis techniques, e.g. Fault-Tree Analysis and Event-Tree Analysis. *(N.B. A role for traceability within ARP 4754 is shown in figure 2.)*

It can be seen therefore that stakeholders involved in the development of SCS - at least according to the above process - require traceability for:- i) the capture and structuring of safety related information produced by the FHA, PSSA and SSA (e.g., hazards, failures, accidents, etc.); ii) analysis towards preparation of a safety case; and iii) enforcing consistency across this information. Once such mechanisms are in place, change control and evolutionary development can be supported.

2.2 Change Control and Impact Analysis

Handling change is a costly and difficult problem; e.g. the loss of Ariane V was partly blamed on ESA's failure to propagate new requirements to all parts of the design [9]. The main problem is the so-called *ripple-effect* where changes to one artifact can have an unforeseen impact elsewhere in the system (e.g., changes to fundamental aircraft-level requirements can propagate to system-level requirements and potentially, requirements at several subsidiary layers of hardware and software). To manage its effects and maintain confidence in a system, certain change related information must

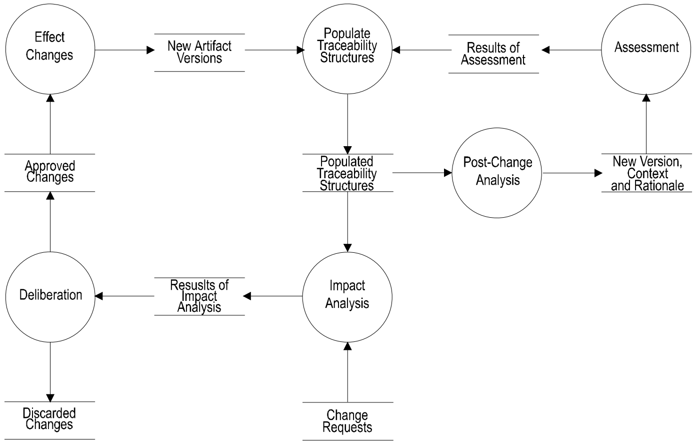


Fig. 3. Change Control Process

be established. Traceability can be an enabling mechanism in this respect and hence, a powerful ally in helping control change (its role in change control/impact analysis is shown in figure 3).

2.3 Evolution of Artifacts

Reuse of requirements and design components in new product developments is an acceptable means of reducing time-to-market without impairing dependability. Airbus, e.g., based an entire range of aircraft around two standard fuselage sections that can be stretched or shortened; their A319 is a variant of its fore-runner, the A320, with system changes confined to the adaptation of software to accommodate the different handling characteristics of a shorter fuselage (this *serial* form of evolutionary development is shown in figure 4).

Evolutionary development can also occur in *parallel* as variants of the same product; e.g., the A340-500 and A340-600 respectively, were jointly developed as higher capacity and extended range derivatives of the A340.

Whilst the economic and technical benefits of reuse are clear, approaches have often been informal, opportunistic and based largely on engineers accumulated knowledge of past projects. Besides the missed opportunities, this has raised questions over safety (reuse encourages the propagation of errors). Traceability is therefore fundamental to the practice of evolutionary development: i) it enables a more systematic approach to assessing reuse candidature of artifacts; and ii) following re-deployment, it facilitates the tracking of artifacts to 2nd, 3rd, 4th, etc. generation variants and conversely, enables the origin of these derivatives to be determined.

In establishing a framework to support these activities, we address a number of questions [10]:-

- **The *scope* of information *coverage*?** Determining the basic information required to support traceability (the ideal being minimal data and maximum utility).

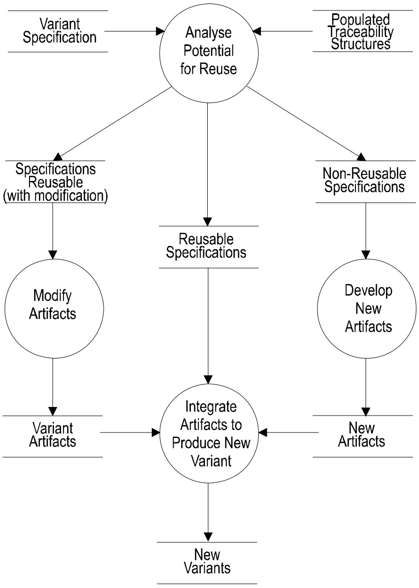

Variant Specification
Analyse Potential for Reuse
Populated Traceability Structures
Specifications Reusable (with modification)
Reusable Specifications
Non-Reusable Specifications
Modify Artifacts
Develop New Artifacts
Variant Artifacts
Integrate Artifacts to Produce New Variant
New Artifacts
New Variants

Fig. 4. Serial Evolutionary Trace Process

- **How to *structure* this information?** Defining traceability structures that group the information into coherent projections reflecting its common usage (i.e. reflecting stakeholder viewpoints).
- **How to *represent* this information?** Means of representing links between traceability artifacts.
- **How to *analyse* this information?** Means of analysing the traceability structures, including verifying the definition and consistency of populated structures and examination/projection of this information.
- **How to implement effective *tool support*?** Means for automating the framework (without it traceability is unsustainable).

Issues raised by the first two questions are seen as technology independent, whereas the rest actually direct the selection of technology.

3 ASTrA: Concepts and Means

This section introduces the theoretical background to our approach. In doing so, we address each of the above questions.

3.1 Traceability Dimensions

To *scope* information coverage, we partition traceability along the following dimensions:-

- *Horizontal Traceability*: traceability between artifacts from the same stage.
- *Vertical Traceability*: traceability between artifacts from different stages.

- *Version Traceability*: traceability between different versions of the same artifact.
- *Variant Traceability*: traceability between artifacts across different projects.

The first three provide dimensions for a cube recording links between project artifacts (figure 5), whilst the fourth relates projects within a product family (figure 6). Figure 5 also illustrates the concepts of *Pre- Requirements Specification* (*RS) Traceability* (between any information source relating to the production of a requirement) and *Post-RS Traceability* (between any information source relating to the development, validation and evolution of a requirement). However, their further discussion is beyond the scope of this paper.

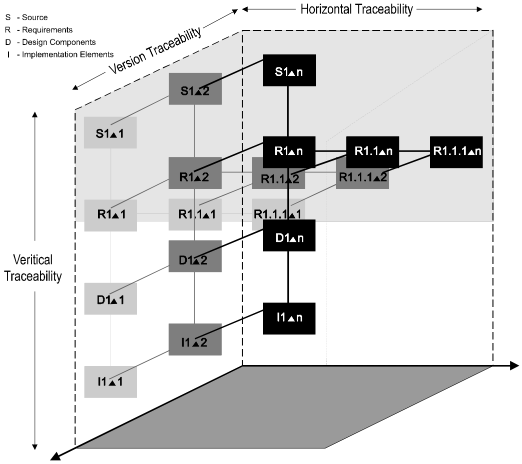


Fig. 5. Traceability of the Horizontal, Vertical & Version Dimensions

These dimensions relate to the traceability concerns for developers of SCS as follows. The safety assessment trace process in figure 2 depicts interactions between the traceability activities (of population and analysis) and the three types of assessment activity identified in ARP 4754; FHA involves tracing between hazards, failures and safety constraints. It therefore demands *horizontal* traceability as the artifacts concerned originate from the same stage (requirements). Conversely, the PSSA and SSA both involve tracing between safety constraints and their strategies for realisation. This demands *vertical* traceability between requirements and design. FHA is conducted at the aircraft and system levels and PSSA at multiple stages of development, from the system level, to hardware and software design. Tracing between assessment results at different levels is therefore another facet of *vertical* traceability.

Conducting impact analysis over assessment artifacts requires support for *horizontal* and *vertical* traceability to determine the ripple effects across these dimensions. For change control, developers further require the ability to populate the *version* dimension with details of revised artifacts. Finally, evolutionary development requires the ability to relate safety properties between projects and to effect impact analysis/change control over these artifacts; i.e., it requires *variant* traceability.

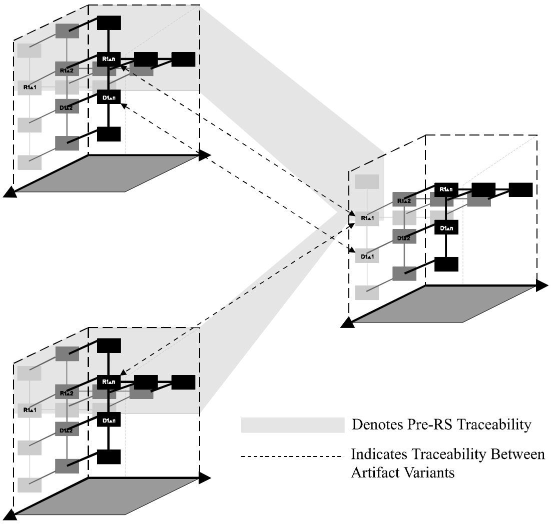

R1▲n
R1▲2
R1▲1
D1▲n
D1▲2
R1▲n
R1▲2
R1▲1
D1▲n
D1▲2
D1▲1
R1▲n
R1▲2
R1▲1
D1▲n
D1▲2
Denotes Pre-RS Traceability
Indicates Traceability Between Artifact Variants

Fig. 6. Traceability of the Variant Dimension

3.2 Traceability Structures

An effective means of recording development and assessment information to support different stakeholder needs is required. Most approaches employ a single structure to represent several (often) disjoint concepts [11]. Given the scope of information required for SCS, this is unsuitable. The alternative is to provide an integrated set of (self-contained) structures, each projecting a different view across the system engineering process. This is approach is favoured by [12] and [13] whose early work on the Design Rationale Capture System (DRCS) underpins our approach.

The DRCS comprises a set of traceability structures (capable of adaptation to specific domains) which focus on particular aspects of development information. Of the five component structures of DRCS, we are concerned with the *artifact synthesis* (records the basic structure of a system) and *argumentation* (records the reasons for or against an assertion). We extend the artifact synthesis here to model *artifact failures* based on Laprie's notion of a fault pathology [14].

3.3 Axiomatic Representation of Traceability Structures

To illustrate our approach, we define a semantics for a Fault Tree based on AND gate and OR gate connectives. Set theory and logic are used to specify its structure (*representation*) and define inference rules and structural constraints for navigation (*analysis*). Approaches to formalisation of Fault Trees have focused mainly on support for the assessment of particular formal models of a system [15], rather than providing a structure to record results of an assessment.

3.3.1 Entity Sets and Relations (Base Axioms)

The following define primitive entity sets for a Fault Tree comprising AND/OR connectives:-

F denotes a finite set of faults: $\{ft_1, ..., ft_n\}$

GAND denotes a relation of ordered pairs $\in$ F, P(F) connected via AND gates: $GAND \subseteq F \times P(F)$

GOR denotes a relation of ordered pairs $\in$ F connected by OR gates: $GOR \subseteq F \times F$

R denotes a set of fault connectives: $GAND \cup GOR$.

FT denotes a Fault Tree: (F, R).

3.3.2 Deductive Axioms Defining Transitive Relationships

Given the base axioms GAND and GOR, we can define rules for causality within a Fault Tree. A basic causal relationship (cause) between a fault (ft) and a set of faults (fs) can be expressed as follows:-

$$(ft, fs) \in \mathit{cause} \Leftrightarrow (ft, fs) \in \mathit{GAND} \vee \forall\, ft' \in fs\text{: } (ft, ft') \in \mathit{GOR} \qquad (1)$$

The transitive closure is defined as:-

$$(ft, fs) \in \mathit{trans\_cause} \Leftrightarrow$$
$$(ft, fs) \in \mathit{cause} \vee \exists\, fs' \subset F\text{: } (ft, fs') \in \mathit{cause} \wedge \exists\, fs'' \subset F, \exists\, ft' \in fs'\text{:}$$
$$(ft', fs'') \in \mathit{trans\_cause} \wedge (fs' - \{ft'\}) \cup fs'' = fs \qquad (2)$$

Trans_cause can be further refined by introducing 'basic events' (BE) of F (failures not further refined by GAND or GOR):-

$$(ft, fs) \in \mathit{cut\_set} \Leftrightarrow (ft, fs) \in \mathit{trans\_cause} \wedge fs \subseteq \mathit{BE} \qquad (3)$$

When ft represents a top event, the relation cut_set defines the minimal cut sets.

3.3.3 Deductive Axioms for Structural Consistency & Completeness Constraints

It is also necessary to formally specify syntactic constraints governing the structure of a Fault Tree; e.g., a well defined Fault Tree must prevent cyclical relationships (and hence ensure termination). This can be expressed as:-

$$\mathit{no\_cycles} \bullet \forall\, fs \subset F, ft \in F\text{: } \mathit{trans\_cause}\,(ft, fs\ ) \Rightarrow ft \notin fs \qquad (4)$$

Similarly, the following constraint prevents a parent node from being linked to child nodes via both AND gate and OR gate connectives:-

$$\mathrm{Dom}(\mathit{GAND}) \cap \mathrm{Dom}(\mathit{GOR}) = \emptyset \qquad (5)$$

In addition, various 'style rules' can be defined to further constrain the structure of a Fault Tree; e.g., we can specify a constraint indicating that no single failure leads to some top event (i.e., where the Fault Tree *cut set* comprises just one event). This can be formally stated as follows:-

$$ft \in \mathit{no\_single\_failure} \bullet (ft, fs) \in \mathit{cut\_set} \Rightarrow |fs| > 1 \qquad (6)$$

Finally, a constraint to detect occurrences of common-cause failures; i.e., events influencing more than one higher event:-

$$\text{ft, ft'} \in \textit{no\_common\_cause} \bullet$$
$$\forall \text{ fs, fs'} \subset F\text{: } \textit{trans\_cause} \text{ (ft, fs)} \wedge \textit{trans\_cause} \text{ (ft', fs')} \Rightarrow \text{fs} \cap \text{fs'} = \varnothing \qquad (7)$$

This model could be extended to include quantitative aspects of FTA by, e.g., associating probabilities with each failure event and identifying inference rules to calculate top event probabilities.

3.4 Tool Support

An OODDB is used to encode the axiomatic representations of the traceability structures. OODDBs combine mathematical logic (where the database can be viewed as a set of base and deductive axioms), with facilities for representing complex structural and behavioural information. We begin by defining a *Fault* class. This is to be used in providing a full description of each *Fault*:-

```
Class Fault with attribute label : String end
```

Sets of faults are represented as follows:-

```
Class FaultSet with attribute fault : Fault end
```

GAND and GOR relationships are modelled as classes. For an OR gate, this can be defined using the Fault class as:-

```
Class Gor with attribute output: Fault; input: Fault end
```

A *Gand* relation is similar, although its *input* is specified in terms of a *FaultSet*:-

```
Class Gand with attribute output: Fault; input: FaultSet end
```

Constraints have been defined to maintain the structural integrity of a Fault Tree, such as *uniqueoutput* and *multiinput* for *Gand*:-

```
Gand with constraint uniqueoutput : $(forall f1,f2/Fault ga/Gand (ga
output f1) and (ga output f2) ==> IDENTICAL (f1,f2))$;
multiinput : $(forall fs/FaultSet ga/Gand
(ga input fs) ==> (exists f1,f2/Fault (fs fault f1)
 and (fs fault f2) and not IDENTICAL (f1,f2)))$ end
```

The *uniqueoutput* constraint safeguards consistency when specifying *output* for a *Gand* class, whilst *multiinput* ensures that *input* cannot be singular, and that they are different. Finaly, the previous constraint preventing a parent node from being linked to child nodes via both AND gate and OR gate connectives can be defined in our database as follows:

```
Gand with constraint domain : $(forall f1,f2/Fault go/Gor ga/Gand (go
output f1) and (ga output f2) ==> not IDENTICAL (f1,f2))$end
```

Having encoded all of the featured structures, inference rules were defined to support analysis over them, e.g., impact analysis and the preparation of a safety case (omitted here for reasons of space).

4 Application of Approach

This section illustrates our approach using a case study of the Wheel Brake System (WBS) for a hypothetical aircraft (described in the ARP 4761). We demonstrate how the results of a FTA are integrated with traceability structures from the other development and assessment activities. We also relate our approach to the traceability dimensions.

4.1 Case Study: Aircraft Braking System

The WBS is installed on the aircraft main landing gears. Its purpose is to decelerate the aircraft on the ground without skidding the tyres. It performs this function automatically upon landing or manually upon pilot activation. We demonstrate our approach using a composite structure (figure 7) which features a subset of the structures and includes an aggregation hierarchy of a WBS. This is represented as an *artifact synthesis* structure (component 'A') and would typically be populated as part of the development process.

Our approach to traceability is applied in the context of the safety assessment process of ARP4754 (figure 2). A FHA has identified the failure behaviour 'unable to decelerate wheels on the ground' (during the phases of landing or rejected take off), a condition caused by total loss of the wheel braking function. A safety constraint has also been imposed which states that loss of all wheel braking (unannunciated or annunciated) shall be less probable than 5E-7 per flight. This constraint is derived in part from knowledge and experience of WBS failure conditions and by conducting a preliminary FTA, from the top event 'loss of aircraft' provides, a set of (system level) failure conditions and associated requirements; tracing between the safety constraint and this top event constitutes pre-RS traceability (see figure 5).

Details of hazards, accidents and failures, together with system constraints (plus rationale for claims a particular constraint excludes a hazard) established through FHA are used to partially populate the *safety specification* and a*rtifact failure* structures (components 'B' and 'C'), and to populate an *argumentation* structure (component 'D1'). These in turn, provide inputs to the PSSA.

A PSSA based on ARP standards typically employs several safety analysis techniques, including Fault Trees which are developed for all critical failures. Figure 8 depicts the (qualitative) results of such an analysis. The top event to the fault tree is 'unannunciated loss of all wheel braking' which is caused by the 'loss of annunciation capability' and 'loss of all wheel braking capability'. We focus on 'loss of all wheel braking', which is caused by 'failure of the normal, alternate and emergency braking systems'. A cause of 'normal brake system does not operate' is 'loss of the Brake System Control Unit (BSCU) ability to command braking'. The two BSCU computers monitor various signals denoting critical aircraft and system states to generate correct braking functions. Loss of this ability is caused by 'loss of power to both BSCU systems', or by a 'BSCU fault' resulting from simultaneous failure of BSCUs 1 and 2.

Information emerging from a PSSA is recorded in the structures; i.e., results of a FTA are used to populate a *fault tree* structure (component 'E') and to complete the a*rtifact failure* structure. Further, a strategy for realising the constraint to exclude 'high speed overrun' (using multi-mode braking and dual BSCUs), plus reasons supporting this claim, contribute to completion of the *safety specification* structure and to population of an *argumentation* structure (composite component 'D2') respectively. Quantitative analysis of the Fault Tree would assign failure rates to basic events and show the top event has a probability of < 5E-7. The *claim* element of 'D2' would be 'loss of all wheel braking is less than 5E-7 per flight', with the *justification* element instantiated as a reference to the Fault Tree (linked via the *has-result* relationship).

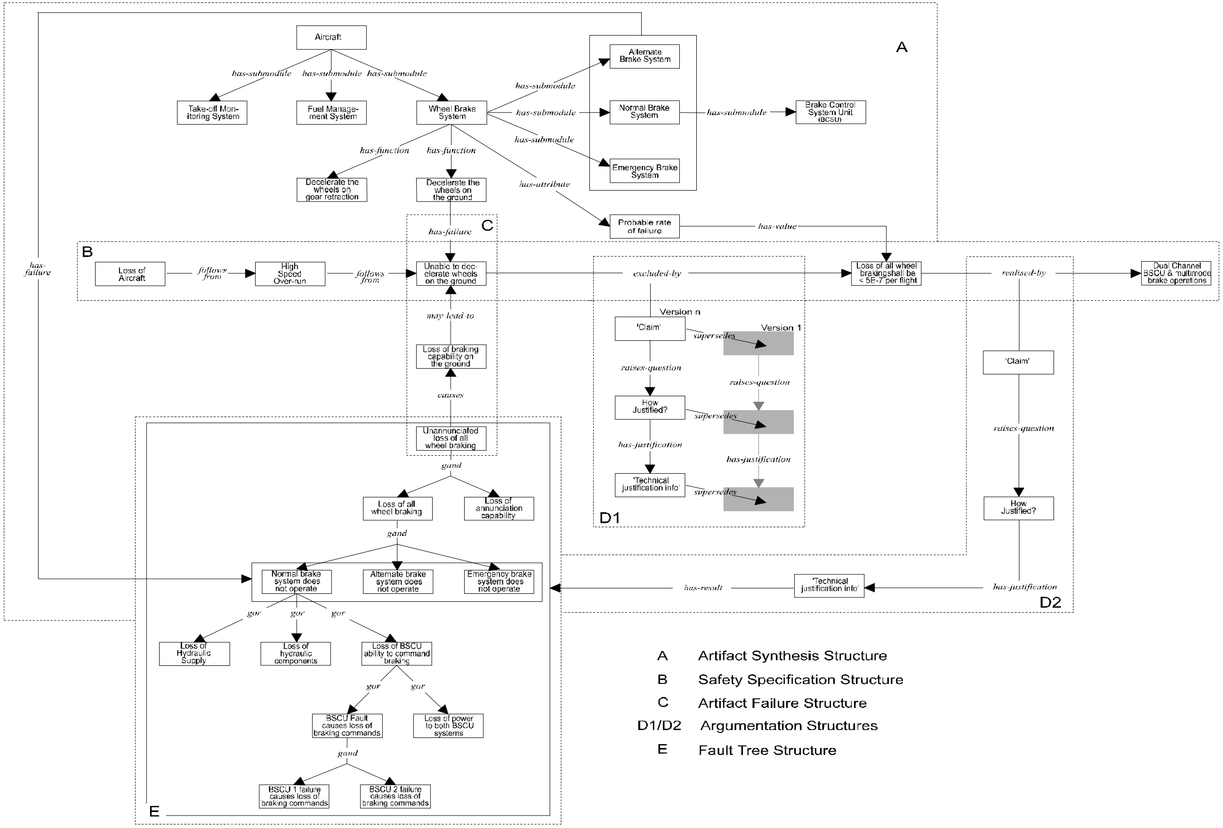

A
Aircraft
has-submodule
has-submodule
has-submodule
Take-off Monitoring System
Fuel Management System
Wheel Brake System
has-submodule
has-submodule
has-submodule
Alternate Brake System
Normal Brake System
Emergency Brake System
has-submodule
Brake Control System Unit (BCSU)
has-function
has-function
has-attribute
Decelerate the wheels on gear retraction
Decelerate the wheels on the ground
Probable rate of failure
has-value
C
has-failure
B
Loss of Aircraft
follows from
High Speed Over-run
follows from
Unable to decelerate wheels on the ground
excluded-by
Loss of all wheel brakingshall be < 5E-7 per flight
realised-by
Dual Channel BSCU & multimode brake operations
may-lead-to
Loss of braking capability on the ground
causes
Unannunciated loss of all wheel braking
has-failure
Version n
Version 1
'Claim'
supersedes
raises-question
raises-question
How Justified?
supersedes
has-justification
has-justification
'Technical justification info'
supersedes
D1
'Claim'
raises-question
How Justified?
has-justification
'Technical justification info'
has-result
D2
gand
Loss of all wheel braking
Loss of annunciation capability
gand
Normal brake system does not operate
Alternate brake system does not operate
Emergency brake system does not operate
gor
gor
gor
Loss of Hydraulic Supply
Loss of hydraulic components
Loss of BSCU ability to command braking
gor
gor
BSCU Fault causes loss of braking commands
Loss of power to both BSCU systems
gand
BSCU 1 failure causes loss of braking commands
BSCU 2 failure causes loss of braking commands
E
A Artifact Synthesis Structure
B Safety Specification Structure
C Artifact Failure Structure
D1/D2 Argumentation Structures
E Fault Tree Structure

Fig. 7. Composite Structure for Tracing Safety Properties

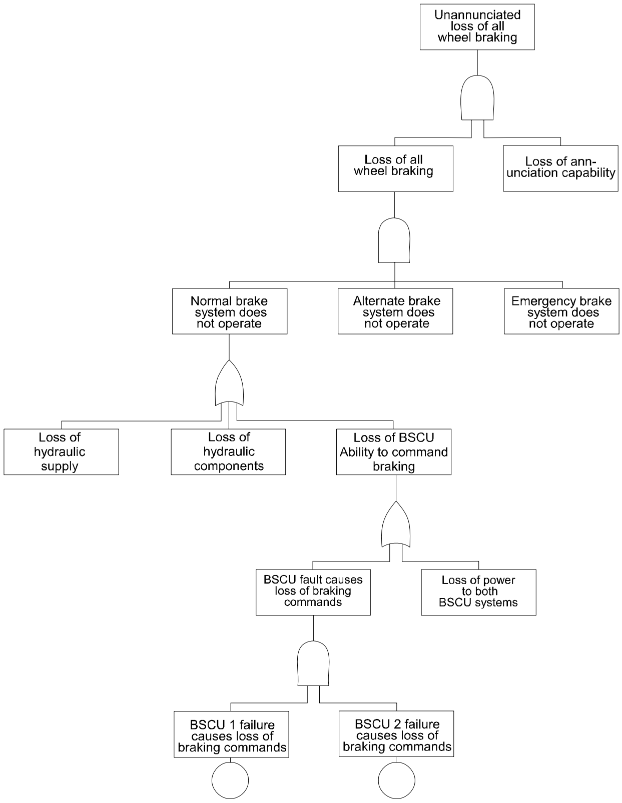

Unannunciated loss of all wheel braking
Loss of all wheel braking
Loss of annunciation capability
Normal brake system does not operate
Alternate brake system does not operate
Emergency brake system does not operate
Loss of hydraulic supply
Loss of hydraulic components
Loss of BSCU Ability to command braking
BSCU fault causes loss of braking commands
Loss of power to both BSCU systems
BSCU 1 failure causes loss of braking commands
BSCU 2 failure causes loss of braking commands

Fig. 8. Wheel Braking System Fault Tree

Following detailed design, a SSA would be developed to support certification. This adds to the analysis, information unavailable during PSSA, e.g., replacing budgeted Fault Tree probabilities with actual values. Relating to our example, this process would typically begin as a bottom up analysis at the item level, with an FMEA of the BSCU power supply. It would be followed by further assessment, including a Zonal Analysis of the main landing gear bay itself. Further traceability structures would be used to support this activity and record the information to emerge.

The safety assessment trace process (figure 2) applied here is seen as cyclical in that changes arising from the SSA feed back into development activities. The composite could be extended to include structures representing other safety analysis techniques, as well as structures capturing, e.g., responsibilities and tasks involved in realising a strategy, exploration of design alternatives, etc.

4.2 Relating ASTrA to the Dimensions for Traceability

Support for *horizontal* traceability is illustrated by the relationship between *accident* ('loss of aircraft'), *hazard* ('high speed overrun'), *failure* ('unable to decelerate wheels on the ground') and *constraint* ('loss of all wheel braking shall be < 5E-7') which originates from the requirements stage.

Support for *vertical* traceability is demonstrated by the relationship between the above *constraint* and its *strategy* for realisation ('dual channel BSCU and multi-mode brake operations), a cross stage relationship between requirements and design. The relationship between system level *failure* and 'BSCU failure' likewise constitutes vertical traceability, since the Fault Tree top event relates to violation of a requirement and basic events to failures in the design components that discharge it.

The approach outlined here can be further extended to support both *version* and *variant* traceability. Indeed, component D1 of figure 7 illustrates basic version traceability in terms of the *supersedes* relationship for an argumentation structure; this would be useful where for example, a claim and its supporting arguments no longer hold and further rationalisation is necessary.

5 Concluding Remarks

This paper has presented an axiomatic approach to traceability for SCS in the context of the ARP 4754 standard. In doing so, it considered key issues in realising traceability, i.e., *scope* of information coverage, its *structure*, *representation* and *analysis*, together with *tool support*. For *scope*, four dimensions (horizontal, vertical, version and variant) to guide stakeholders in determining which elements to trace were proposed. Using case study material, this information was *recorded* in an integrated set of traceability structures, each relating to a particular view of the development and assessment process. The representation of a FTA as a traceability structure and its relation to structures from other development and assessment activities was also shown; e.g., in figure 7, assessment is applied over a development artifact via *has-failure*, whilst the products of assessment support relationships between development artifacts; *has-result* being a case in point. Base and deductive axioms were then shown to provide appropriate means of *representing* and *analysing* the structures. Constraints were also defined over a Fault Tree. Finally, we demonstrated the potential for *tool support* by encoding these axioms in an OODDB.

Most traceability approaches have no formal basis, exceptions being [16] and [17]. Inference and assertion have also been used to represent subjective forms of information such as decision rationale and non-functional requirements [18, 19] Regarding safety analysis, Adelard's Safety Case Editor [20] employs means for tracing safety properties, but is weak in non-safety related areas.

References

1. Arkley, P., Riddle, S.: Overcoming the Traceability Benefit Problem. In: Procs. Int. Conf. on Requirements Engineering (2005)
2. Sharif, B., Maletic, J.: Using Fine-Grained Differencing to Evolve Traceability Links. In: Procs. IWESE, USA, March 22-23 (2007)
3. Cert'n. Considerations for Highly-Integrated or Complex Aircraft Systems, ARP 4754 (1996)
4. EUROCAE, Software Considerations in Airborne Systems & Equipment (1992)
5. Mason, P.: Extending Cross-Tool Traceability to Formal Methods. In: Procs. of 2nd Int. Conf. on Advances in Information Technology, Bangkok, Thailand (November 2007)

6. Mason, P.: Extending Cross-Tool Traceability to Source Code. In: Procs. of ECTI-CON Conference, Thailand (May 2007)
7. Mason, P., Tianvorakoon, A.: Seamless Traceability without Compromise. In: Procs. Advances in Computer Science and Technology, Phuket, Thailand (April 2007)
8. Health & Safety Commission, The Ladbroke Grove Rail Inquiry (2001)
9. Lions, J.: ARIANE 5 Failure Report, ESA (1996)
10. Pohl, K.: Process-Centered Requirements Engineering. John Wiley and Sons, Chichester (1996)
11. Berg, K., Bishop, J.: Tracing Software Product Line Variability. In: Procs. of SAICSIT, pp. 111–120 (2005)
12. Mohan, K., Ramesh, B.: Tracing Variations in Software Product Families. CACM 50(12) (2007)
13. Klein, M.: Capturing Design Rationale in Concurrent Engineering Teams. IEEE Computer, 39–47 (January 1993)
14. Avizienis, A., Laprie, J., Randell, B.: Fundamental Concepts of Computer System Dependability. In: Workshop on Robot Dependability, Seoul, Korea (May 2001)
15. Mason, P.: A Rigorous Object-based Specification of Fault Trees. In: Procs. 6th JCSSE, Thailand (2009)
16. Spanoudakis, G., Zisman, A., Perez-Minana, E., Krause, P.: Rule-based Generation of Requirements Traceability Relations. Journal of Systems & Software 72(2) (2004)
17. Avila Garcez, A.S., Russo, A., Nuseibeh, B., Kramer, J.: Combining Abductive Reasoning & Inductive Learning to Evolve Requirements Specifications. IEE Procs. Software (February 2003)
18. Mylopoulos, J., Chung, L., Nixon, B.: Representing and Using Non-functional Requirements: A Process Oriented Approach. IEEE Trans. Soft. Eng. 6(18) (1992)
19. Maurer, F.: Project Co-ordination in Design Processes. In: Proc. 5th Workshop on Enabling Technologies, USA, pp. 191–198 (June 1996)
20. Emmet, L., Guerra, S.: Application of a Commercial Assurance Case Tool to Support Software Certification. In: Procs. Software Certificate Manag't Workshop (2005)

Token Traversal Strategies of a Distributed Spanning Forest Algorithm in Mobile Ad Hoc - Delay Tolerant Networks

Apivadee Piyatumrong[1], Patricia Ruiz[1], Pascal Bouvry[1], Frederic Guinand[2], and Kittichai Lavangnananda[3]

[1] Faculty of Science, Technology & Communication, University of Luxembourg
{apivadee.piyatumrong,patricia.ruiz,pascal.bouvry}@uni.lu
[2] Le Havre University, France
frederic.guinand@univ-lehavre.fr
[3] School of Information Technology, Bangkok, Thailand
kitt@sit.kmutt.ac.th

Abstract. This paper presents three distributed and decentralized strategies used for token traversal in spanning forest over Mobile Ad Hoc Delay Tolerant Networks. Such networks are characterized by behaviors like disappearance of mobile devices, connection disruptions, network partitioning, etc. Techniques based on tree topologies are well known for increasing the efficiency of network protocols and/or applications, such as Dynamicity Aware - Graph Relabeling System (DA-GRS). One of the main features of these tree based topologies is the existence of a token traversing in every tree. The use of tokens enables the creation and maintenance of spanning trees in dynamic environments. Subsequently, managing tree-based backbones relies heavily on the token behavior. An efficient and optimal token traversal can highly impact the design of the tree and its usage. In this article, we present a comparison of three distributed and decentralized techniques available for token management, which are Randomness, TABU and Depth First Search.

Keywords: Token traversal, spanning tree, distributed system, delay tolerant networks, Depth First Search.

1 Introduction

Networks spontaneously and automatically created between neighboring mobile devices are commonly called ad hoc networks. The main advantage of this kind of networks is that no infrastructure or administration system is required. The signal strength can be weakened due to the appearance and disappearance of the devices, the mobility of nodes, and obstacles in the environment. These phenomenons lead to frequent and long duration partitions of the network. An emerging subclass of ad hoc networks, Mobile Ad Hoc Delay Tolerant networks (henceforth called *mobile ad hoc DTNs* for brevity), is characterized by including these undesirable behaviors. This unpredictable and highly fluctuating topology makes challenging many aspects like efficient communication, routing, etc.

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 96–109, 2009.

Previous researchers demonstrated the validity of spanning trees in networking area [1], [2], etc. Establishing a spanning tree in the network is a well known prerequisite strategy for providing efficient communication and routing algorithms in wired networks. Furthermore, recently it is also a tendency to use them in mobile ad hoc DTNs [3, 4, 5]. One common mechanism used in spanning tree algorithms is the utilization of tokens. In [6], the authors state that techniques for traversing the token that perform well in static networks are not necessarily well suited in networks with high mobility. Thus, a new study of token traversal in high mobility network must be undertaken. Also in [7], it is concluded that the token movement strategies impact on the tree construction, and, therefore, on the topology management. This motivated us to study, implement and compare different token traversal techniques in order to determine which strategy performs better in different environments in mobile ad hoc DTNs.

Dynamicity Aware - Graph Relabeling System (DA-GRS) [8] is a model for creating and analyzing decentralized topologies and algorithms targeting dynamically distributed environments like mobile ad hoc DTNs. Up to now, the token traversal strategies used in DA-GRS are based on the assumption that no memory is used in the mobile nodes. These techniques are random and Tabu [7].

In this study, we applied Depth First Search (DFS) for the first time to DA-GRS. We considered to include DFS in our comparison since it is a very well known strategy for static tree traversal and the idea of DFS has been used for mobile ad hoc networks in recent works [9]. However, due to the highly fluctuant topology, having an ordering strategy might not be a good idea. Thus, a deep study and also a comparison between DFS and other techniques are needed.

This study assumes that spanning trees provide a reliable path way for efficient communications and services. Thus, having a spanning tree covering as many nodes as possible in the shortest time is desired. In the context of this study, the spanning tree must span the entire connected communication graph. Therefore, we implemented and compared different strategies for traversing the token in the tree topology in terms of the performance ratio and the convergence speed rate. The performance ratio is measured as the number of different partitions (or connected components) of the underlying network divided by the number of existing trees. The convergence speed rate shows how fast multiple trees belonging to the same partition merge into one tree. We compare three different distributed strategies: Randomness, TABU, and Depth First Search (DFS), described later in Section 4.

The contribution of this paper is thus three-fold: (1) the design and implementation of DFS for DA-GRS for the first time and the re-implementation of other two token traversal techniques, (2) implementation of a framework which allows different token movements strategies based on realistic mobility models (e.g. highway and shopping mall scenarios), and (3) a comprehensive study and analysis of the three proposed strategies which is currently missing in the literature.

The paper is organized as follows; Section 2 introduces the conceptual rules for constructing and maintaining a spanning forest in mobile ad hoc DTNs in a purely distributed and decentralized manner. Later, in Section 3, the realistic communication models used for creating the tree topology over an existing mobile ad hoc DTNs are explained. After that, in Section 4 all the compared strategies for circulating the token are presented. The experiments realized in this paper are explained in Section 5 and the results obtained are shown in Section 6. Finally, Section 7 concludes the work.

2 The Spanning Forest Algorithm Using DA-GRS Model

Dynamicity Aware - Graph Relabeling System (DA-GRS) [8] is an extension of Graph Relabeling System, GRS [10]. It is a high level abstraction model that can improve the development of self-organized systems. All the mechanisms underlying it are for managing mobile ad hoc DTNs efficiently. DA-GRS just models how to handle with topology changes and interaction between devices, but it does not itself create services or applications.

A tree is defined as a free cycle graph. A graph composed of several trees is called, hereinafter, a forest. Assuming this, DA-GRS proposes some rules for constructing and maintaining a spanning forest in mobile ad hoc DTNs represented in Figure 1 and described after. In this figure, the circle represents a node. Letters on top of the nodes mean: (1) 'T' if the node possesses the token, (2) 'N' if the node does not possess the token, and (3) 'Any' when the node can possess or not the token. The labels '0', '1' and '2' on the edge represent the route to the token. And finally, label 'off' describes a broken link.

Dynamic networks are characterized by mobility and possible connection disruptions, hence, devices need to handle with this changes when creating and maintaining the spanning tree. DA-GRS proposed four rules (as shown in Figure 1) to handle with four different situations. In the initial state every device has the token (what means it is a tree itself), and these 4 rules are:

- rule 1: A tree link breaks, and the node belongs to the sub-tree which does not possess the token. In this case the node must regenerate the token, otherwise there will exist a tree without a token (which is an undesirable situation).
- rule 2: A tree link breaks, and the broken link occurs at a node which currently belongs to the sub-tree which possesses the token. In this case, the node does nothing regarding the maintenance of the token.
- rule 3: When a node with token meets another device possessing a token; both nodes will try to merge their trees in order to obtain a bigger tree from the two existing

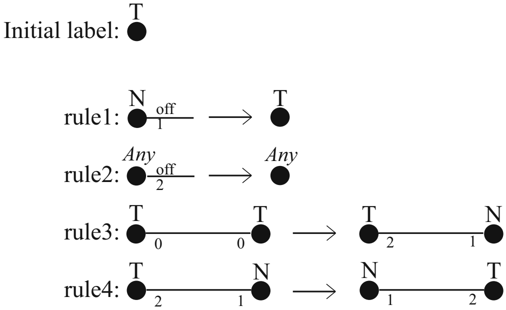


Fig. 1. DA-GRS rules for creating and maintaining spanning forest topologies

ones. The trees merging process starts. The result of this rule remains a bigger tree and only one token (the merging process discards one token automatically in order to remain one and only one token within a tree).
- rule 4: Token traversal in general case: the token visits the nodes of the tree following a given strategy.

An important feature of this model is that in each tree one and only one token exists. Furthermore, only two nodes possessing token (thus belonging to different trees) can start the trees merging process. As we are dealing with trees, cycles are not allowed. DA-GRS manages to avoid them since it is not possible to have two nodes belonging to the same tree and possessing a token at the same time.

3 Utilizing and Applying DA-GRS for Creating the Spanning Forest

For creating a spanning forest over a mobile ad hoc DTN using DA-GRS in a decentralized way, nodes must exchange some messages between them in order to have knowledge of who else is possessing a token in the neighborhood and also to merge trees. Since no global knowledge is considered, a more detailed communication syntax needs to be specified. Therefore, the proposed message sequence that devices must exchange in a decentralized system is explained in the following:

3.1 Beaconing

In order to have knowledge of the one-hop neighborhood most decentralized systems utilize beacons (also called ‘hello messages’) [11]. For that purpose, every node sends periodically a message alerting about its presence. For considering a node as a neighbor, one must receive a beacon of the node regularly. A node will not be a neighbor anymore when its beacon is not received within a predefined time.

Using this beaconing both a broken communication link and the appearance of a new one-hop neighbor are detected, and thus, ‘rule 1’ and ‘rule 2’ in Figure 1 can be applied. Based on *Beaconing Rate* of IEEE802.11 [12], the time interval used for periodically sending the beacon is 100 millisecond.

3.2 Trees Merging Process

‘rule 3’ in Figure 1 represents the spanning tree construction scenario (trees merging process). DA-GRS uses rendez-vous assumption [13] as synchronization method at this merging process. This rendez-vous assumption states that at one moment in time, only two nodes possessing token can meet and be merged. We consider that this assumption is too rigid in real world communications. Thus, this work proposes to relax this assumption by allowing a node to choose one token among the tokens owned by its neighbors.

In a distributed system a node has no ability to know if there exists any node with token in its neighborhood. Thus, nodes holding a token will broadcast a packet, *‘findingTk’*, to verify whether any of its neighbors also possesses a token. If any neighbor of this

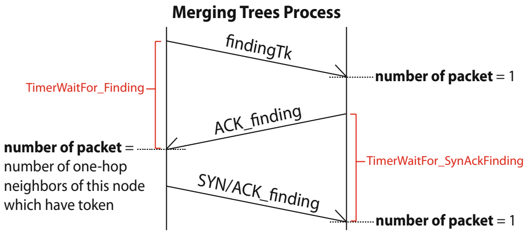


Fig. 2. Message sequence diagram for merging trees

broadcasting node possesses a token and receives *'findingTk'* will reply using a *'ACK_finding'* message. *'ACK_finding'* is an expression of agreement to merge their trees.

Moreover, this particular neighbor will set its status to wait for *'SYN/ACK_finding'* to confirm the merging process within a predefined period, *'TimerWaitFor_SynAckFinding'*. As we are working with a discrete simulator, the time duration of the timers is one simulation step.

After broadcasting *'findingTk'*, the broadcasting node will wait within a predefined duration, *'TimerWaitFor_Finding'*. At the end of this waiting time, the broadcasting node selects one of its neighbor and a *'SYN/ACK_finding'* message will be sent using unicast to this selected neighbor. In case, there is no node with token in the neighborhood, at the end of this timer the token is circulated. The message sequence of this process is illustrated in Figure 2.

3.3 Token Traversal

'rule 4' in Figure 1 stands for token traversal in general case. When a node sends a broadcast message for finding a neighbor possessing a token, it also establishes a timer as addressed in previous section, *'TimerWaitFor_Finding'*. If the timer finishes and there is no answer from any neighbor, the token movement takes place. If there is no neighbor belonging to a different tree, the node will directly move the token, see Figure 3.

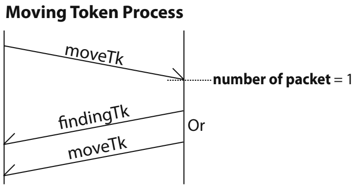


Fig. 3. Message sequence diagram for traversing the token

4 Token Traversal Strategies in a Decentralized System

As explained in previous sections, in DA-GRS and usually when dealing with spanning trees, the system need a token for creating and maintaining a tree. Every node, at some moment, must possess the token, since it allows looking for neighbors with token to merge trees. The way this token moves along the tree impacts on the spanning tree construction. In literature, tree traversal refers to the process of visiting each node in a tree data structure in a particular manner [14]. In the context of this study, we want the token to traverse less but has more chance to meet another token. In other words, we want the fastest rate of the tree construction to cover a connected subgraph, which means less number of trees or in the best case, remaining only one tree over a connected subgraph. In previous work, the token traversal in DA-GRS has only been implemented using random or Tabu techniques. This is the first time that DFS is applied to DA-GRS.

This section gives a detailed explanation of the three strategies used in this study. It is worth noting that all strategies are working in distributed and decentralized manner suiting to work in mobile ad hoc DTNs.

4.1 Randomness

The Randomness here follows the uniform distribution law. Randomness is the heuristic used by DA-GRS by default. The process is done by selecting a node randomly among the list of neighbors. The description of the 'Randomness' traversal technique is described in Algorithm 1.

Algorithm 1. Using Randomness heuristic in $Move\_Token$ (τ_i) process of a node ν

```
1: $\alpha$ is the set of neighbors of node $\nu$
2: node $\rho$ is a node selected randomly from set $\alpha$
3: move token $\tau_i$ from node $\nu$ to node $\rho$
```

4.2 TABU

TABU creates a list of forbidden movements in which the most recent nodes possessing the token are stored. This list is called as tabu list. The algorithm consults the tabu list before sending the token to a neighbor in order to avoid visiting the same node repeatedly. Tabu list uses a fix size of memory, *memory_size*, to set the number of stored nodes in the list. This list is sent within the token, no node memory is used. In Algorithm 2 a detailed description of this strategy is given.

Applying this technique to DA-GRS was proposed in previous work [7]. The *memory_size* of the list (its length) was also studied in [7], and it was demonstrated that a tabu list longer than 1 entry of device did not provide much better results than using a tabu list with size 1. Therefore, we use in our study TABU with size list equal to 1. For brevity, henceforth we will use 'TABU$\{1\}$' to represent the usage of TABU at *'memory_size'* equals to one. This is equivalent to prohibiting sending the token to the node from which the current one received it.

4.3 Depth First Search (DFS)

DFS is commonly used as token movement technique [5, 15, 16] when dealing with tree based topologies. It imitates the traversal of the classical Depth First Search algorithm and, thus, it is an ordering traversal strategy.

Algorithm 2. Using TABU heuristic in $Move\_Token$ (τ_i) using a defined value of *memory_size* processing at a node ν

```
$\alpha$ is the set of neighbors of node $\nu$
$\beta$ is the TABU-like list which has size equal to memory_size
Set $availableNode$ = $\alpha$ - $\beta$
if $availableNode \neq \emptyset$ then
    node $\rho$ is a node selected randomly from set $availableNode$
    token $\tau_i$ move from node $\nu$ to node $\rho$
    if the number of item of $\beta$ reach the memory_size then
        remove the first item from list $\beta$
        add $\rho$ to the end of list $\beta$
    else
        add $\rho$ to the end of list $\beta$
    end if
else
    node $\rho$ is a node selected randomly from set $\alpha$
    remove item $\rho$ from list $\beta$
    token $\tau_i$ move from node $\nu$ to node $\rho$
    add $\rho$ to the end of list $\beta$
end if
```

In order to traverse systematically like the classical algorithm in distributed and dynamic systems, DFS utilizes the neighbor list information provided by the beaconing process. Thus, the neighbor list is always up to date. Furthermore, in this implementation, it is necessary to keep information inside each node. To be more specific, these information are: (a) about the node that sends the token to the current device for the first time (henceforth, we refer to this first node as 'upper neighbor'), and (b) information of neighbors receiving the token from this current device. In this way, the node will definitely sends the token to all its neighbors using the neighbor list and the information stored (a) and (b). It will not send the token back to the upper neighbor meanwhile all the list of neighbors is not visited.

The mechanism is as follows: whenever the current node receives the token back from its neighbors (and this is not the first time this node receives token), the current node will send the token to the next neighbor in the neighbor list. Once the list is finished, the token is sent back to the 'upper neighbor' if it has not gone from the neighborhood. Otherwise, this current node will become its own 'upper neighbor' and will send again the token to the first neighbor of the its neighbor list. This implementation is described in Algorithm 3.

5 Experiment Methodology and Measurements

5.1 Experiment Methodology

The networks used in this work were generated using a discrete network simulator, Madhoc [17]. An ad hoc networks simulator that provides mobility models allowing realistic motion of citizens in variety of environments. Two real-world mobility models,

Algorithm 3. Using DFS heuristic in $Move\_Token$ (τ_i) process of a node ν

```
$\alpha$ is the set of neighborhood of node $\nu$
$\beta$ is the DFS list in node $\nu$
$\varpi$ is '$upper\ neighbor'$'
$\delta$ is the latest node that send $\tau_i$ to $\nu$
if $\varpi$ is empty then
    $\varpi = \delta$
end if
Set $availableNode$ = $\alpha$ - $\beta$ - $\varpi$
if $availableNode \neq \emptyset$ then
    node $\rho$ is the first node from set $availableNode$
    move token $\tau_i$ from node $\nu$ to node $\rho$
    add $\rho$ to the end of list $\beta$
else
    clear list $\beta$
    if $\varpi$ is in the set $\alpha$ then
        move token $\tau_i$ from node $\nu$ to node $\varpi$
        set $\varpi$ to empty
    else
        $\varpi = \nu$
        Set $availableNode$ = $\alpha$ - $\delta$
        node $\rho$ is the first node from set $availableNode$
        move token $\tau_i$ from node $\nu$ to node $\rho$
        add $\rho$ to the end of list $\beta$
    end if
end if
```

'shopping mall' and 'highway', were selected in the simulations using the parameters summarized in Table 1.

Mobile ad hoc DTNs can be represented as a dynamic communication graph (G), where the mobile devices are the set of vertices (V), and the links between them are the edges of the graph, (E). The dynamism of the network is represented by the fact that

Table 1. Parameters used in the experiments

	Shopping Mall	High way
Surface (km^2)	0.32	2.24
Node density (per km^2)	347.85	72.55
Number of nodes	110	160
Avg. Number of partitions	1.95	15.9
Number of connections	446	498
Average degrees	8.13	6.23
Velocity of nodes (m/s)	0.3-3	20-40
Radio transmission range	40-80 m	
Network technology	IEEE 802.11b	

both V and E can change at any time. Therefore, the graph at a given time t, $G(t)$, is composed of $(V(G(t)), E(G(t)))$.

We derived communication graphs from Madhoc which performs simulation in discrete-time. So the communication network corresponds to a series of static graphs: $G(t)$ for $t \in \{t_1, t_2, t_3, ..., t_{400}\}$. Between two consecutive times t_i and t_{i+1} the communication graph remains the same. However, using such a short timing-snapshot, 1/4 seconds between two consecutive times is considered sufficient to reflect the reality. We made 100 runs for each experiment in order to have reliable results. That means we are simulating the behavior of a mobile ad hoc DTN for 100 seconds over 100 different topologies.

5.2 performanceRatio() Function

At a given moment t, $G(t)$ may be partitioned into a set of m connected subgraphs. Having Γ as the set of all spanning trees at moment t of $G(t)$. The quality of the algorithms can be assessed by the number of connected subgraphs (m) over number of trees created (Γ). This quality is determined by the following ratio.

$$performanceRatio(G(t)) = \left(\frac{m}{\mid \Gamma \mid}\right) \tag{1}$$

The value of the performance ratio approaching to one means higher quality of the algorithm (less number of trees in a connected subgraph). Having a spanning tree per connected subgraph enables more efficient communication and topology management, since at least, the information can be disseminated systematically via the created spanning tree. This means the algorithm is robust regarding the dynamism of the network because it can construct a tree covering all the nodes conforming the connected subgraph.

Figures 4(a) and (b) illustrate the measurement of all cost functions proposed here. In the figure 4(a), the communication graph $I(t)$ has two connected subgraphs, and each connected subgraph has one spanning tree. On the contrary, the communication graph $K(t)$ depicted in figure 4(b) has only one connected subgraph but four spanning

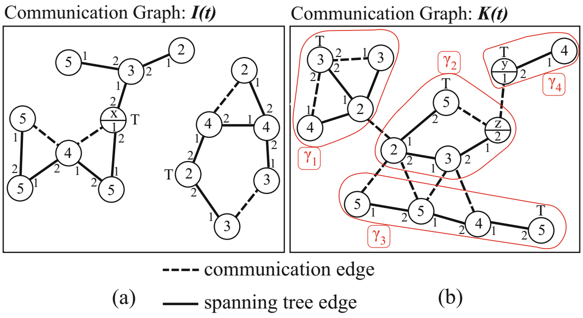


Fig. 4. An example scenario for illustrating the proposed cost functions for spanning forest

trees ($\gamma_1, ..., \gamma_4$). Thus, the $performanceRatio(I(t))$ and $(K(t))$ equal to 1 and 0.25, respectively.

5.3 convergenceSpeedRate() Function

The $convergenceSpeedRate()$ is measured based on the number of iterations in simulation. Let Δ be the number of iterations the algorithm required trying to achieve the least $performanceRatio()$ and Δ^* be the number of iterations required per $G(t)$. Having $performanceRatio()$ equal to one within $G(t)$ is an ideal situation. However, having limited merging process (explained in Section 3) causes no guarante that $performanceRatio()$ will be one, in other words, it is always possible to have multiple trees per connected component at any time t of graph G. In such case, the number of iterations used within that $G(t)$ will be counted into Δ. The lower the value of $convergenceSpeedRate()$ is, the faster the algorithm converges a connected component into a tree. The $convergenceSpeedRate()$ can be written as below.

$$convergenceSpeedRate(G(t)) = \left(\frac{\Delta(G(t))}{\Delta^*(G(t))}\right) * 100 \qquad (2)$$

6 Results

In this section we present the comparison results obtained for the three different strategies studied for circulating the token in a decentralized tree based algorithm. These three strategies are: Randomness, TABU$\{1\}$, and DFS. The comparison was made in terms of the speed of the convergence of the tree and the performance ratio explained both in the previous section. The results shown are the average of simulating 100 topologies for 100 seconds each topology.

Figure 5 and 6 show the simulation results for the shopping mall and highway environment, respectively. From both figures, DFS clearly gives the best behavior among these three strategies for both environments. Furthermore, both figures show the impact of the mobility model toward the resulting tree. Easily observing from Figure 5 (shopping mall scenario), the difference between DFS and the other strategies is very large. Contrary, the resulting gap from DFS to other strategies in the highway scenario, Figure 6, is not as big as what we can see in shopping mall environment. This is because of the speed of the devices and hence, the highly fluctuant topology. Thus, we measure the differences between DFS and the other two strategies (averaging results over the simulation time). The results of this measurement are shown in Table 2 to demonstrate the difference in terms of the percentile of the distance. Furthermore, as the result values do not follow a normal distribution in any case, we apply the Kruskal-Wallis test in order to obtain statistical significance with 95% probability in our comparisons. The results show that DFS are significantly better than TABU$\{1\}$ and Randomness.

According to Table 2, the differences are up to 60-70% when compare the result of DFS and the other strategies in shopping mall model. On the other hand, for highway model, the differences of results between all the strategies are distinguishable and also statistically significant, but not so huge different (12-35% of differences) as found in

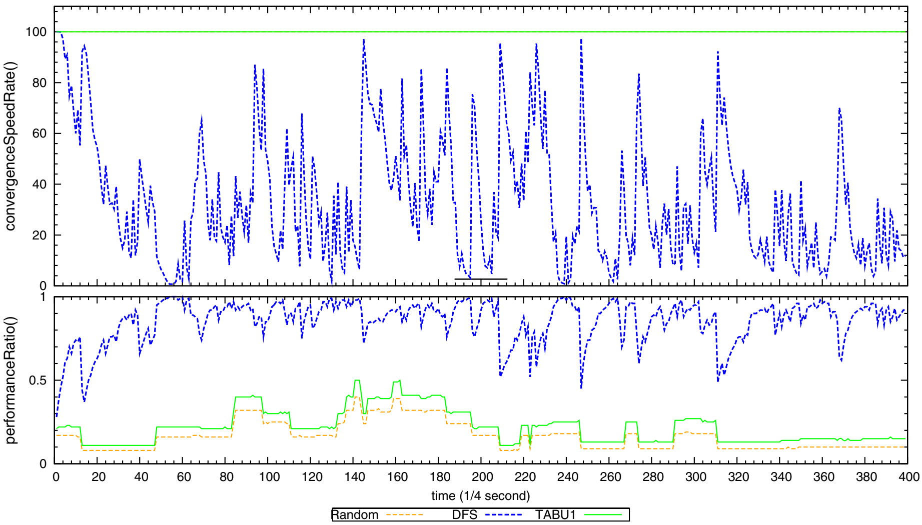


Fig. 5. Comparison of convergenceSpeedRate() measuring among all studied algorithms in 'Shopping Mall' mobility model

Table 2. The percentile of the distance from DFS to the other strategies

		TABU{1}	Randomness
Highway	performanceRatio	23.71%	34.71%
	convergenceSpeed	12.38%	12.40%
Mall	performanceRatio	63.70%	69.19%
	convergenceSpeed	66.42%	66.42%

shopping mall model. This comes from the fact that the highway model has a high fluctuating mobility. Thus, the topology is more likely to change than in the shopping mall.

The overall results show that the Randomness strategy is the worst one in both environments. This behavior was expected, since when using the random technique many nodes in the tree can hardly possess the token, so the merging trees in those areas rarely happened. TABU{1} ensures that one neighbor will not possess the token twice consecutively. Thus, TABU{1} achieves better distribution of the token than Randomness.

As stated at the beginning of this paper, our intuition suggest that a strict ordering strategy may not be a promising technique for a high changeable topology. The reason to this intuition is that the topology is changing a lot in a short time while the token is moving mostly in the same area during such small period of time. However, the experimentation results deny our intuition. Our results show that DFS behaves better than Randomness and TABU{1}. Thus, it can be confirmed that an ordering strategy like DFS can work well under highly changing topology.

In Table 3 we resume the behavior of each technique since we can consider this is a multi-objective multi-constraint problem, and depending on the necessities of each

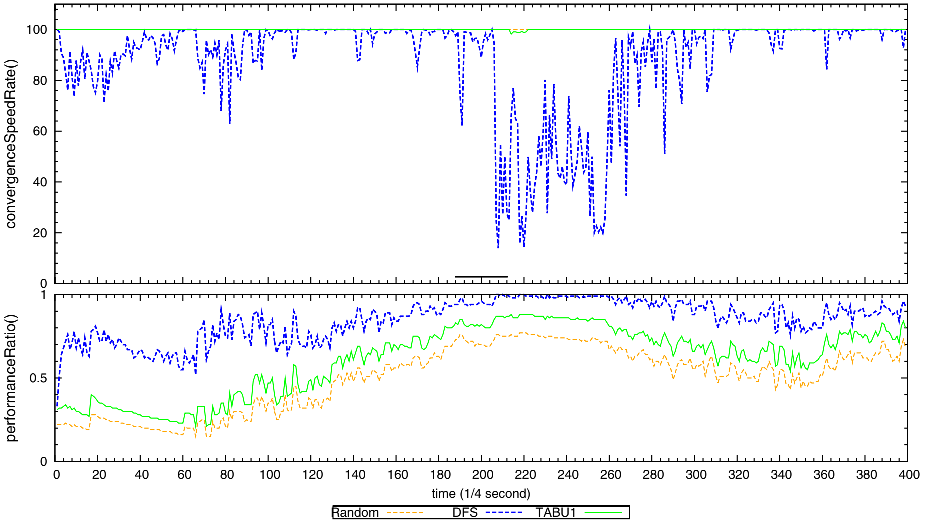


Fig. 6. Comparison of convergenceSpeedRate() measuring among all studied algorithms in ‘highway’ mobility models

Table 3. Multi-objective multi-constraint study

	DFS	TABU	Randomness
No memory	No	No	Yes
Token memory	No	Yes	No
Node Memory	Yes	No	No

situation a technique or another can be used. As it can be seen in Table 3 the only strategy that uses no memory at all is Randomness. For those using memory it is possible to distinguish between using the memory in the node like DFS, or using memory in the token as TABU.

7 Conclusions and Future Work

Providing efficient communication and topology management in delay tolerant mobile ad hoc networks is a difficult task which presents a real challenge. We found out that token traversal techniques generally used in tree-based algorithms has a significant impact to the resulting tree. In this work, three different strategies for token movement through the tree topology: Randomness, TABU$\{1\}$, and Depth First Search (DFS) were systematically studied and compared in terms of the performance ratio and the speed of convergence. The former measures the number of spanning trees per connected component at a given moment, the closer to one the better performance. The latter gives an idea of how fast different trees belonging to the same connected component merge and form a solely tree composed of all the nodes within the partition.

To the best of our knowledge, it is the first time DFS is applied to DA-GRS, a mobile ad hoc DTN system, and also it is the first time a comparison between token traversal techniques is done in the literature.

For doing the comparison, two different scenarios were selected: (1) a shopping mall where the movement of the device is slow, and (2) a highway where the nodes move at high speeds. We found out that ordering strategies for token traversal helps to merge trees faster. This can be confirmed since DFS outperform the no ordering techniques like Randomness and less ordering such as TABU$\{1\}$.

As future work, we plan to study the impact of these techniques to any high level application when using the tree, i.e., when disseminating a message through the whole network using this tree based topology, routing, etc. Since the token movement affects the creation of the tree, therefore we also want to study how these strategies impact on the robustness of application using tree-based topology.

References

[1] John, E.C.R., McQuillan, M., Richer, I.: The new routing algorithm for the arpanet. IEEE Transactions on Communications COM-28(5), 711–719 (1980)

[2] IEEE standard 802.1d-2004: IEEE standard for local and metropolitan area networks: Media access control (mac) brideges (June 2004)

[3] Daly, E.M., Haahr, M.: Social network analysis for routing in disconnected delay-tolerant manets. In: MobiHoc 2007: Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing, pp. 32–40. ACM, New York (2007)

[4] Su, J., Goel, A., de Lara, E.: An empirical evaluation of the student-net delay tolerant network. Annual International Conference on Mobile and Ubiquitous Systems, 1–10 (2006)

[5] Ruiz, P., Dorronsoro, B., Khadraoui, D., Bouvry, P.: BODYF–A Parameterless Broadcasting Protocol Over Dynamic Forest. In: Workshop on Optimization Issues in Grid and Parallel Computing Environments, part of the High Performance Computing and Simulation Conference (HPCS), pp. 297–303 (2008)

[6] Malpani, N., Chen, Y., Welch, J.L.: Distributed token circulation in mobile ad hoc networks. IEEE Transactions on Mobile Computing 4(2), 154–165 (2005)

[7] Pigné, Y.: Modélisation et traitement décentralisé des graphes dynamiques - application aux réseaux mobiles ad hoc. Ph.D. dissertation, L'Université du Harve (December 2008)

[8] Casteigts, A.: Model driven capabilities of the DA-GRS model. ICAS 2006: Proceedings of the International Conference on Autonomic and Autonomous Systems, 24 (2006)

[9] Gandhi, R., Mishra, A., Parthasarathy, S.: Minimizing broadcast latency and redundancy in ad hoc networks. IEEE/ACM Transactions on Networking 16, 840–851 (2008)

[10] Sopena, E., Metivier, Y.: Graph relabeling systems: a general overview. Computers and Artificial Intelligence 16(2), 167–185 (1997)

[11] Gast, M.: 802.11 Wireless Networks: The Definitive Guide, 2nd edn. O'Reilly, Sebastopol (2005)

[12] IEEE standard 802.11: Wireless lan medium access control and physical layer specifications, August 1999. IEEE Computer Society, Los Alamitos (1999)

[13] Cardelli, L.: An implementation model of rendezvous communication. LNCS, vol. 197. Springer, Heidelberg (1985)

[14] Goodaire, E.G., Parmenter, M.M.: Discrete mathematics with graph theory, 2nd edn. Prientice Hall Inc., Upper Saddle River (2002)

[15] Bauer, N., Colagrosso, M., Camp, T.: An agile approach to distributed information dissemination in mobile ad hoc networks. In: IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Washington, DC, USA, pp. 131–141 (2005)

[16] Stojmenovic, I., Russell, M., Vukojevic, B.: Depth first search and location based localized routing and qos routing in wireless networks. In: Intl. Conf. on Parallel Processing (ICPP 2000) (2000)

[17] Hogie, L., Bouvry, P., Guinand, F., Danoy, G., Alba, E.: Simulating Realistic Mobility Models for Large Heterogeneous MANETS. In: Demo proceeding of the 9th ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM 2006), pp. 129–141. IEEE, Los Alamitos (2006)

Requirements for a Knowledge Transfer Framework in the Field of Software Development Process Management for Executive Information Systems in the Telecommunications Industry

Nalinpat Porrawatpreyakorn[1], Gerald Quirchmayr[1,2], and Wichian Chutimaskul[3]

[1] University of Vienna,
Faculty of Computer Science, Department of Distributed and Multimedia Systems,
Liebiggasse 4, A – 1010 Vienna, Austria
a0848231@unet.univie.ac.at, Gerald.Quirchmayr@univie.ac.at
[2] University of South Australia, School of Computer and Information Science,
Mawson Lakes, SA 5095, Australia
Gerald.Quirchmayr@unisa.edu.au
[3] King Mongkut's University of Technology Thonburi,
School of Information Technology,
126 Prachauthit Rd., Thungkru, Bangkok, 10140, Thailand
wichian@sit.kmutt.ac.th

Abstract. This paper presents the interview findings of the current situation in software development process management for Executive Information Systems (EIS) in the Thai telecommunications industry and identifies requisite requirements for a successful Knowledge Transfer (KT) framework that consists of two proposed components, which are frameworks themselves: the proposed Software Development Life Cycle (SDLC) maintenance framework and the proposed KT framework. The resulting frameworks are aimed at providing an improved way of software development process management to better achieve the high performance goals of successful software development.

Keywords: SDLC Maintenance Framework, Knowledge Transfer Framework, Executive Information Systems, Telecommunications Industry.

1 Introduction

The use of EIS has significantly increased since the success of EIS in many developed countries stimulates a number of executives to adapt this Information Technology (IT) into their organizations in order to compete in an increasingly competitive environment. EIS can be defined as computer-based Information System (IS) that supports communications, coordination, planning and control functions of managers and executives in an organization. The data provided by EIS are taken from many different internal and external sources. Executives make most, if not all, decisions based on a mainly qualitative thought process. Consequently, soft information is critical in executive decision making [1].

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 110–122, 2009.

The characteristics of EIS include an easy to use and maintainable graphical user interface; integrated capability for data access, security, and control; on request "drill down" capability to lower levels of details; depiction of organizational health indicators; functionality for decision support, ad hoc queries, and what-if analysis; data analysis features; advanced report generation; statistical analysis tools for summarizing and structuring data; and access to a variety of external data sources [2].

EIS are different from Transaction Processing Systems (TPS), Management Information Systems (MIS), and Decision Support Systems (DSS) in terms of problems addressed, users, and data used. TPS serve operational management by performing and recording the daily routine transactions necessary to conduct the business and solve structured problems. Both MIS and DSS provide middle management. However, there are different characteristics for the way in which an MIS deals with summarized and compressed data from the TPS and sometimes does an analysis of that summarized data in order to solve structured problems, while DSS use data from TPS, MIS, and external sources to solve semi-structured problems. An EIS provides information for top management to solve unstructured problems so that they can identify problems and opportunities by combining information from both, internal and external sources [3, 4].

Regarding the development of new software systems, an SDLC model and a project management concept are necessary. SDLC is a tight combination of software development phases and process model. The best known approach still is the waterfall approach, which in fact is the oldest fully developed SDLC. It is a systematic and sequential pattern. It typically begins with an initial feasibility study through to maintenance of the IS. Although most of organizations use it for managing EIS development projects, there are several limitations. For example, Jirachiefpattana [4] describes that these limitations consist of well-defined requirements, lack of flexibility to improve initial imperfections in one phase, time-consuming, too much documentation and high cost. It thus is important to use an appropriate SDLC framework for dealing with such disadvantages. Besides, the project management concept is employed for planning, organizing and managing resources to bring about the successful development. Albeit a number of IS development methodologies are proposed, these methodologies are mostly tested within western countries and considered as universal rules [5]. Hence for effective software development in particular types of projects and particular areas, the models intended for use need to be geared towards meting the specific requirements and have to be tested accordingly to suit the current situation in the software development.

At present, most business environments are quickly changing and increasingly competitive. Competitive markets can increase the industry's economic efficiency, provide more choices for consumers, encourage innovation and the use of the most effective methods of production, and boost growth [6, 7]. Telecommunications still is one of the most rapidly evolving competitive markets and one of the fastest-growing areas of technology in the world. As this paper focuses on the Thai telecommunications industry, Thailand's telecommunications sector it is worth mentioning that it has continued to grow, in the last five years. One sector of the market that has been particularly popular is broadband internet. Thailand was one of five Asian countries ranked among the world's top ten fastest-growing consumer broadband markets in 2007 [8]. Unfortunately, software developers are still under tremendous pressure to deliver quality results in order to effectively respond to these highly competitive environments. In the context of the Thai telecommunications industry, very typical problems of an EIS development process management, such as a lack of effective ways to manage the EIS development and to transfer knowledge and experience within the EIS development teams, are faced by a majority of software developers. This cause influentially leads EIS development to sometimes ineffective and unsatisfactory results. As this paper aims at software development with better performance, it is intended to lead to significant software development achievements and business achievements; it therefore should be of great interest for both software developers and telecommunication companies.

Consequently, the answers of the questions of "how to construct an SDLC maintenance framework for EIS and how to organize KT during EIS development" should help EIS development teams by acting as guidance to increase EIS development effectiveness and efficiency.

2 The Current Situation in the Software Development Process Management for EIS in the Thai Telecommunications Industry

For getting an idea of the current situation in the EIS development process management in the Thai telecommunications (by focusing on Internet services), we use findings of interviews with software development teams working for two companies: True Corporation Public Company Limited and TOT Public Company Limited. They are the two biggest broadband Internet Service Providers (ISP) in the Bangkok region and have their own optical fiber cable networks in Bangkok and in the vicinity. Even though there are many ISP in Thailand, most of them still lease bandwidth from one of these two companies. With 85% True Corporate also has the largest market share [9, 10]. In the Thai organizational context of EIS implementation, the results reveal that the executives could sometimes not provide adequate participation in the projects. Subordinates did not have full authorities when it came to making decisions. Communication processes in organizations also were quite complicated. These limitations resulted in development teams sometimes not being able to effectively identify the information requirements from executives, often having to wait for Steering Committee decisions, and resulted in an extensive organizational process. Finally, as was to be expected, the projects were delayed.

Given the underlying EIS development strategies, the EIS development project with a small team in True Corporation had a short duration. True Corporation used a prototyping model. The development process involved requirements analysis, preliminary design, prototype design, construction and testing, implementation and maintenance. In the other case, the EIS development project with a big team in TOT had an initial period of two years. Outsourcing usually covers a wide range of contractual arrangements ranging from contract programmers to third party facilities management [11]. The EIS development in TOT was to some extent outsourced. The formal reason for employing the consultant was that the internal staff lacked knowledge and experience in EIS development. A development methodology supplied by the consultant was used for the EIS development. Although the methodology used terms like prototyping and module delivery, it can best be characterized as a variant of the waterfall approach. The development process involved a large execution of requirements analysis, system development, user acceptance, system installation, use and maintenance.

During EIS development, the teams face similar problems. For example, users between business units do neither have good cooperation nor do they participate well; users provide inadequate requirements specifications and quite frequently change their requirements; users have only limited IT/IS skills; and so on. This situation is quite typical, and not limited to the Thai telecommunications sector. KT processes are also practised in a very similar way, such as by discussing and sharing ideas in regular meetings; transferring theoretical knowledge by self-learning that is based on the existing internal documents; providing practical training case by case during EIS development; and finally supporting theoretical training prior to project for specialists. Consequently, this contributes to a rather small amount of theoretical knowledge transfer. Based on the four categories of failure factors of Chow and Cao [12], the problems/failures of these findings can be summarized in Table 1. However, this is an assumption that we have not yet been able to empirically verify with real data.

Table 1. The failure factors in EIS development

Dimension	Factor
Organization	Lack of management commitment, organizational culture too political, lack of agile logistical arrangement
People	Lack of necessary skill-set, lack of project management competence, lack of good user participation and cooperation between business units, lack of team work
Process	Ill-defined project scope, requirements, and planning, user team having no full authority
Technique	Lack of provision and support of appropriate training to team, inappropriateness of methods and tools

3 The Current Focus in Europe

The main purpose of this study is to improve performance of software development process management for EIS. Since European countries such as the UK, France, Nordic and Germanic countries are recognized for success in EIS and software development, this study focuses on two main parts in Europe. For the first focus on software development management with success factors, especially agile methods are most commonly adopted and have generated lots of interest in the software development industry. Comparing between SDLC models, Schwaber [13] concludes that Scrum has more advantages than others in facets of responsiveness to environment, team flexibility and creativity, knowledge transfer during project, and high probability of success. Furthermore, for covering all aspects of project management in the traditional sense, Fitsilis [14] suggests that connecting Scrum with PMBOK can benefit the software project management since processes in Scrum and PMBOK, are addressed in a compatible way. In addition, defined success factors should be oriented towards completing software development effectively. Based on prior studies on the factors that impact on agile software projects, these identified important factors can be summarized in Table 2.

Table 2. A summary of the identified success factors of agile software projects

Source	Factor
[15]	Individual competence, management support, communication, compatibility of agile methods, teamwork, project type, team size
[16]	Individual competence, compatibility between skills and tasks, good communication skills, experience in software development
[17]	Resistance due to past experience, micromanagement (leadership), career consequences, developers' ability
[18]	External support, teamwork, compatibility of methods, negotiation skills
[19]	Teamwork, individual ability, motivation
[20]	Organizational culture, management style, organizational form, management of software development knowledge, reward systems, teamwork, competence, user relationships, existing technology and tools, training

For the second focus on the KT process during software development with success factors, in context of software development, where it has multiple stakeholders with varying backgrounds and knowledge, KT is vital to effective software development success. Communication plays a crucial role in KT process. Joshi et al. [21] define that the KT process is collectively determined by five components: source context, recipient context, knowledge context, relational context, and situational context. The source and recipient context refer to the attributes of the knowledge source and recipients which can facilitate or impede the process of knowledge transfer. The relational context refers to the attributes that characterize the relationship between a knowledge source and a recipient. The knowledge context refers to the nature and characterization of the type of knowledge that is being transferred. The situational context refers to the environmental characteristics surrounding the knowledge transfer process. Additionally, based on prior literature concerning the influential factors that affect KT effectiveness, these identified factors can be summarized in Table 3.

Table 3. A summary of the identified success factors of KT process

Source	Factor
[21]	Great motivation, source's capability, source's credibility, knowledge type, good relationship, extensive communication
[22]	Great motivation, source's capability, source's credibility, receipt's absorptive capacity and retentive capacity, knowledge type, good relationship, extensive communication
[23]	Source's capability, source's credibility, extensive communication, and culture, knowledge type, good relationship
[24]	Transmission of source's knowledge to potential recipient and absorption of the knowledge by that recipient

In conclusion, there is an amount of IS research that focuses on the improvement of software development success in aspects of speed, effectiveness, efficacy, and low cost, to only name the most important ones. It is commonly accepted that no single method can serve for all types of software development projects and all types of project objectives. Therefore, this study considers agile project management, a process-based KT model, and key requirements for a successful KT framework for EIS development in the Thai telecommunications sector. The setting of this focus was also determined by the scholarship supporting this work.

4 Theoretical Foundations Concerning the Two Proposed Frameworks

The following two sets of principles and three models are used as the basis for proposing a KT framework. First of all, to answer the question of "how to construct an SDLC maintenance framework", the two sets of principles are derived from the core of the PMBOK and the core of the eTOM, which the authors then try to merge into the Scrum model. Next, to answer the question of "how to organize KT during EIS development", our solution is based on Szulanski's KT process model. The two factors of TAM are used as additional key requirements for both of the proposed frameworks.

4.1 Principles of the Project Management Body of Knowledge Guide (PMBOK)

PMBOK developed by the Project Management Institute (PMI) is process-based, meaning that it describes work as being accomplished by processes. The processes are described in terms of inputs, tools and techniques, and outputs. The guide recognizes 42 processes that fall into five basic process groups and nine knowledge areas that are typical of almost all projects. PMBOK is a general guide used by professional project managers to achieve long-term goals and is applied in many software development projects. It also viewed as quasi standard by several leading software development companies [25]. Therefore, PMBOK can be used as input for developing the proposed SDLC maintenance framework.

4.2 Principles of the Enhanced Telecom Operations Map (eTOM)

The Business Process Framework, known as eTOM, is a widely deployed and accepted standard for business processes in the telecommunications industry. The eTOM represents the whole of a Service Provider/Operator's enterprise environment in a hierarchy of process elements that capture process detail at various levels. The framework can be used as a tool for analyzing an organization's existing processes and for developing new processes. Moreover, eTOM provides a basis for understanding and managing portfolios of IT applications in terms of business process requirements and enables the creation of consistent and high-quality end-to-end process flows, with opportunities for cost and performance improvement, and for the reuse of existing processes and systems [26]. Thus, these advantages help to fulfill the requirements for the proposed SDLC maintenance framework without addressing the type of projects in PMBOK.

4.3 The Scrum Model

Scrum, originally developed by Ken Schwaber is an iterative incremental process of software development commonly used in the context of agile software development.

Scrum focuses on project management in situations where it is difficult to plan ahead, with mechanisms for ‘‘empirical process control” and where feedback loops constitute the core element. Software is developed by a self-organizing team in increments (called ‘‘sprints”), starting with planning and ending with a review. Features to be implemented in the system are registered in a backlog. Then, the product owner decides which backlog items should be developed in the following sprint. Team members coordinate their work in a daily stand-up meeting. One team member, the scrum master, is in charge of solving problems that stop the team from working effectively [27].

4.4 Technology Acceptance Model (TAM)

The Technology Acceptance Model (TAM) is an influential extension of Ajzen and Fishbein’s theory of reasoned action (TRA). It was introduced and developed by Fred D. Davis in 1986. TAM is a model derived from a theory that addresses the issue of how users come to accept and use a technology. There are two specific variables, perceived usefulness (PU) and perceived ease of use (PEOU), which are hypothesized to be fundamental determinants of user acceptance of any technology [28, 29]. Usefulness was also considered to significantly influence usage more than ease of use. PU is a strong correlate of user acceptance and should not be ignored by IT/IS development teams attempting to design or implement successful systems [28]. The goal of TAM is to predict IS acceptance and diagnose design problems before users have experience with a system [30]. Additionally, TAM has been applied in numerous studies testing user acceptance of IT/IS, such as, word processors, spreadsheet applications, e-mail, web browser, telemedicine, and websites [31].

4.5 Szulanski’s Knowledge Transfer Process Model

Szulanski’s (1996) theory of a KT process model, as a communication model, describes an intra-firm knowledge transfer. The knowledge transfer process is viewed as a message transmission from a source to a recipient in a given context [32]. In addition, the process follows four stages: Initiation, where the knowledge source distinguishes the knowledge which can meet the destination’s demand; Implementation where both sides construct an appropriate channel for the coming transfer and at the same time the source adjusts the knowledge to adapt to the destination’s demand; Ramp-up where the destination adjusts the received knowledge to make it fit the new setting; and Integration where the destination turns the knowledge into a system of itself and be one part of its own knowledge package [33].

4.6 Towards a Knowledge Transfer Framework

Because the proposed KT framework aims to provide management direction to achieve better performance of telecommunication software development process management, the PMBOK guide, the eTOM framework, the Scrum model, TAM and Szulanski’s KT process model all have strong benefits for the proposed KT framework. In order to specifically answer the first question of “how to construct an SDLC maintenance framework”, the Scrum processes and PMBOK project processes are mapped into five iterative process groups: initiating, planning, executing, controlling, and closing. eTOM is also mapped together by focusing on the top-down principle in order to help decompose processes into component processes to expose more detail, define flows to link processes together, and combine decompositions and flows to fully describe the behavior of each process area. Besides, this study deems critical success factors suitable for the current EIS development situation in the Thai telecommunications industry. Based on the PU and PEOU factors of TAM and identified success factors of the previous studies on agile methodologies, these factors can be categorized into five dimensions: organizational, people, process, technical, and project dimensions.

In order to specifically answer the second question of "how to organize KT during EIS development", Szulanski's KT process model is used with consideration of important factors that contribute to KT effectiveness. These key factors are based on the PU and PEOU factors of TAM and the factor findings of prior literature listed in Table 3 which can then be categorized into five basic components: source context, recipient context, knowledge context, relational context, and situational context. Appropriate guidance of both SDLC maintenance and KT organization during EIS development is vital to achievement of EIS development success with effectiveness and efficiency. Hence to build such appropriate guidance, the requirements for a successful KT framework which are specific to the EIS development in the Thai telecommunications sector need to be identified.

5 Requirements for a Successful Knowledge Transfer Framework

Stating requirements is very important for the design of all mechanisms. Requirements for the successful implementation of the proposed SDLC maintenance framework are summarized into Table 4. This is based on Chow and Cao's [12] dimensions, the PU and PEOU factors of TAM, and the consolidation of a number of failure/success factors listed in Tables 1 and 2 which share similar characteristics.

Table 4. Requirements for the successful proposed SDLC maintenance framework

Dimension	Factor
Organization	Management commitment, organizational environment, team environment
People	Team capability, user involvement, personal characteristics
Process	Project management process
Technique	Agile software technique, perceived usefulness and ease of use, provision and support of training
Project	Project type, team size

Furthermore, Table 5 presents the summary of the requirements for the success of the proposed KT framework. This is also based on the five KT components of Joshi et al. [21], the PU and PEOU factors of TAM, and the consolidation of the problems of the KT process in the current situation in EIS development process management and the defined success factors of prior literature.

Table 5. Requirements for the successful proposed KT framework

Dimension	Requirement	Description
Source context	Ease of use and usefulness	This requirement relates to the understanding of KT process and benefit which gain when transferring KT.
	Great motivation	This requirement relates to motivation to transfer knowledge.
	Capability	This requirement relates to source's greater reservoir of knowledge that has a potential to transfer knowledge.
	Creditability	This requirement relates to source's trust and reputation.
Recipient context	Ease of use and usefulness	This requirement relates to the understanding of KT process and benefit which gain when transferring KT.
	Great motivation	This requirement relates to motivation to absorb knowledge.
	Absorptive capacity	This requirement relates to ability to exploit sources of knowledge.
Knowledge context	Appropriate technical knowledge and project management knowledge	This requirement relates to appropriate knowledge that should transfer to the recipient.
Relational context	Good relationship between team members	This requirement relates to relationship between the source and the recipient since it influentially affect to effective KT.
Situational context	Extensive communication	This requirement relates to extensive communication between the source and the recipient.

6 What a Framework Based on These Requirements Could Look Like

In general, according to [34] each software development project is run through a platform deployment lifecycle of four stages: ideas, feasibility, software development, and rollout. First, the ideas stage starts with a collection of ideas for end user solutions that can be enable through the new software platform. Second, the feasibility study stage is to compile the information needed for the responsible management to make a decision whether to start a pilot development project. Next, the software development stage is where the approved pilot development project is run based on the feasibility study results. Last, the rollout stage runs when developed software is ready to be employed. For this platform deployment lifecycle, most development theories have similar methods for the stages of ideas, feasibility, and rollout. Except for the utilization of software development methodologies, it is dependent on the type, nature, and characteristics of each project. To be clear, the main focus of this study is on the software development stage as presented in Fig. 1. Additionally, Fig. 2 presents the conceptual proposed SDLC maintenance framework based on the identified success factors and the two principles of PMBOK and eTOM, which are then merged into the Scrum model.

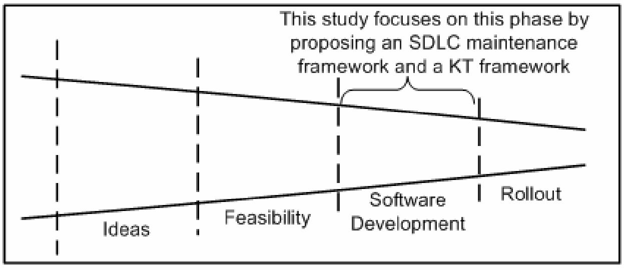

This study focuses on this phase by proposing an SDLC maintenance framework and a KT framework
Ideas
Feasibility
Software Development
Rollout

Fig. 1. The primary focus of this study

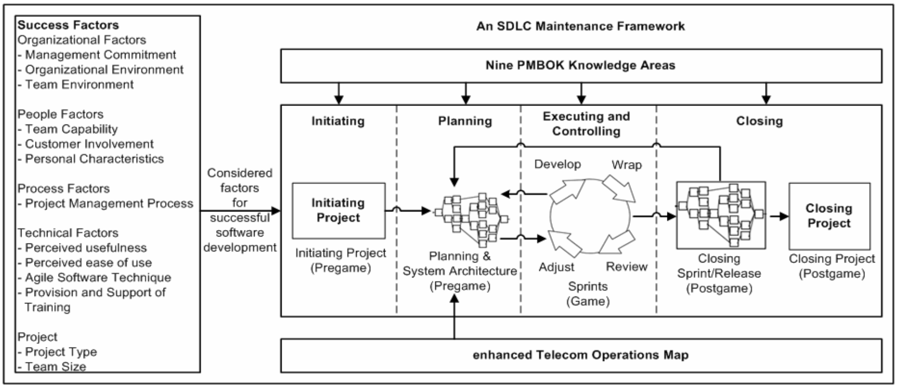

Success Factors
Organizational Factors
- Management Commitment
- Organizational Environment
- Team Environment
People Factors
- Team Capability
- Customer Involvement
- Personal Characteristics
Process Factors
- Project Management Process
Technical Factors
- Perceived usefulness
- Perceived ease of use
- Agile Software Technique
- Provision and Support of Training
Project
- Project Type
- Team Size
Considered factors for successful software development
An SDLC Maintenance Framework
Nine PMBOK Knowledge Areas
Initiating
Planning
Executing and Controlling
Closing
Initiating Project
Initiating Project (Pregame)
Planning & System Architecture (Pregame)
Develop
Wrap
Adjust
Review
Sprints (Game)
Closing Sprint/Release (Postgame)
Closing Project
Closing Project (Postgame)
enhanced Telecom Operations Map

Fig. 2. The proposed conceptual SDLC maintenance framework (Modified from [13])

Fig. 3 presents the conceptual proposed KT framework based on the identified success factors and Szulanski's KT process model.

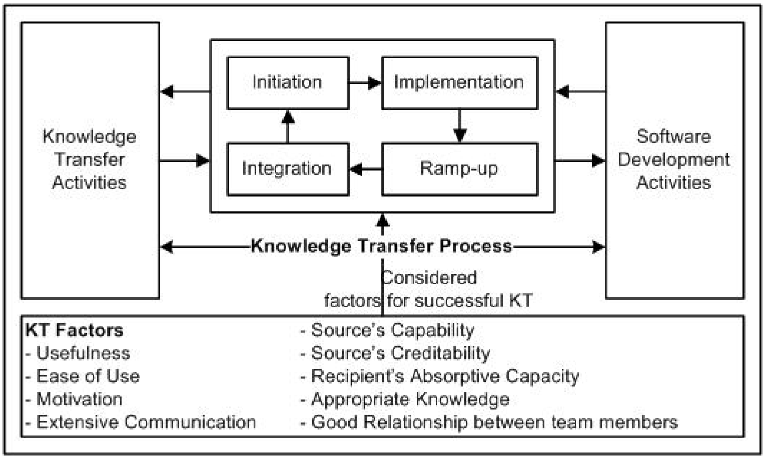

Knowledge Transfer Activities
Initiation
Implementation
Integration
Ramp-up
Software Development Activities
Knowledge Transfer Process
Considered factors for successful KT
KT Factors
- Usefulness
- Ease of Use
- Motivation
- Extensive Communication
- Source's Capability
- Source's Creditability
- Recipient's Absorptive Capacity
- Appropriate Knowledge
- Good Relationship between team members

Fig. 3. The proposed conceptual KT maintenance framework

The key requirements identified for the proposed KT framework serve as basis for designing an abstract level model of the two proposed conceptual frameworks. The next step will be a more detailed process mapping between PMBOK, eTOM, and Scrum for the proposed SDLC maintenance framework. The relationship between crucial factors and KT processes will also be described for the proposed KT framework.

7 Conclusion

The components that play a central role in effective software development success are an SDLC model and a KT concept. In this contribution, the first step to construct the successful SDLC maintenance and KT frameworks is to identify the specific requirements for EIS development in the Thai telecommunications industry. The interview findings show that software developers do not perceive formal routines as an efficient way to manage software development processes and activities, and sometimes to transfer IT/IS knowledge and experience. To deal with this problem, two principle sets of the PMBOK and the eTOM and three models of the Scrum, TAM and Szulanski's KT process all serve as inspirational principles and models for the proposed SDLC maintenance framework and the proposed KT framework. PMBOK provides general guidance covering all facets of project management in traditional sense. eTOM fulfills specific requirements in the area of telecommunications industry. Scrum is commonly used in the agile project management context which suits to the EIS characteristics and nature. The PU and PEOU of TAM are significant factors to the technology acceptance of both proposed frameworks. As communication is at heart of KT, Szulanski's KT process model helps organize KT during software development. The next steps of this work are (1) the detailed process mapping between PMBOK, eTOM, and Scrum for the proposed SDLC maintenance framework and (2) the relationship description between crucial factors and KT processes for the proposed KT framework. Consequently, the proposed framework is aimed at providing an improved way of software development process management with better performance in the field of Thai telecommunications sector.

Acknowledgments. The authors would like to thank ÖAD, the Austrian Agency for International Cooperation in Education and Research, and the Higher Education Commission of Thailand for supporting this work in the form of a scholarship for Nalinpat Porrawatpreyakorn.

References

1. Salmeron, J.L.: EIS success: keys and difficulties in major companies. Technovation 23(1), 35–38 (2003)
2. Nord, J.H., Nord, G.D.: Executive Information Systems: A Study and Comparative Analysis. Information & Management 29(2), 95–106 (1995)
3. Laudon, K.C., Laudon, J.P.: Management Information Systems. Prentice Hall, Upper Saddle River (2009)

4. Jirachiefpattana, W.: The Impact of Thai Culture on Executive Information Systems Development. In: Proceedings of the 6th International Conference Theme 1, Globalization: Impact on and Coping Strategies in Thai Society, pp. 97–110 (1996)
5. Cronholm, S., Ågerfalk, P.J.: On the Concept of Method in Information Systems Development. In: Proceedings of the 22nd Information Systems Research Seminar in Scandinavia, pp. 229–236 (1999)
6. Telecommunications and the Competitive Advantage of Massachusetts, `http://www.law.indiana.edu/fclj/pubs/v47/no2/weld.html`
7. The Donor Committee for Enterprise Development, `http://www.bdsknowledge.org/dyn/be/besearch.details?p_lang=en&p_phase_id=80&p_phase_type_id=2`
8. Point Topic, `http://point-topic.com/content/operatorSource/profiles2/thailand-broadband-overview.htm`
9. The World Bank, `http://siteresources.worldbank.org/INTTHAILAND/Resources/333200-1177475763598/3714275-1234408023295/5826366-1234408105311/chapter4telecommunication-sector.pdf`
10. Thailand Guru, `http://www.thailandguru.com/internet-thailand-adsl-broadband-high-speed.html`
11. Lacity, M.C., Willcocks, L.P.: Global information technology outsourcing: in Search of Business Advantage. Wiley, Chichester (2001)
12. Chow, T., Cao, D.: A survey study of critical success factors in agile software projects. The journal of Systems and Software 81(6), 961–971 (2008)
13. SCRUM Development Process, `http://www.jeffsutherland.org/oopsla/schwapub.pdf`
14. Fitsillis, P.: Comparing PMBOK and Agile Project Management software development processes. In: Sobh, T. (ed.) Advances in Computer and Information Sciences and Engineering, pp. 378–383. Springer, Netherlands (2008)
15. Cockburn, A., Highsmith, J.: Agile software development: the people factor. IEEE Computer 34(11), 131–133 (2001)
16. McManus, J.: Team agility. Computer Bulletin 45(5), 26–27 (2003)
17. Cohn, M., Ford, D.: Introducing an agile process to an organization. IEEE Computer 36(6), 74–78 (2003)
18. Schatz, B., Abdelshafi, I.: Primavera gets agile: a successful transition to agile development. IEEE Software 22(3), 36–42 (2005)
19. Ceschi, M., Sillitti, A., Succi, G., Panfilis, S.D.: Project management in plan-based and agile companies. IEEE Software 22(3), 21–27 (2005)
20. Nerur, S., Mahapatra, R., Mangalaraj, G.: Challenges of migrating to agile methodologies. Communications of the ACM 48(5), 73–78 (2005)
21. Joshi, K.D., Sarker, S., Sarker, S.: Knowledge Transfer Among Face-to-Face Information Systems Development Team Members: Examining the Role of Knowledge, Source, and Relational Context. In: Proceedings of the 37th Hawaii International Conference on System Sciences, pp. 1–11. IEEE Computer Society, Washington (2004)
22. Szulanski, G.: Exploring Internal Stickiness: Impediments to the Transfer of Best practice within the Firm. Strategic Management Journal 17(Winter Special Issue), 27–43 (1996)

23. Sarker, S., Sarker, S., Nicholson, D., Joshi, K.D.: Knowledge Transfer in Virtual Information Systems Development Teams: an Empirical Examination of Key Enablers. In: Proceedings of the 36th Hawaii International Conference on System Sciences, pp. 1–10. IEEE Computer Society, Washington (2003)
24. Davenport, T.H., Prusak, L.: Working with Knowledge: How organizations Manage What They Know. Harvard Business School Press, Boston (2000)
25. PMI Institute: A Guide to the Project Management Body of Knowledge, 4th edn. PMI Institute (2008)
26. TM Forum, `http://www.tmforum.org/BestPracticesStandards/BusinessProcessFramework/1647/Home.html`
27. Schwaber, K., Beedle, M.: Agile Software Development with Scrum. Prentice Hall, Upper Saddle River (2001)
28. Davis, F.D.: Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Quarterly 13, 319–340 (1989)
29. Technology Acceptance Model, `http://www.imresearch.org/RIPs/2005/RIP2005-4.pdf`
30. Dillon, A., Morris, M.G.: User acceptance of new information technology: theories and model. Annual Review of Information Science and Technology 31, 3–32 (1996)
31. Applying the Technology Acceptance Model (TAM) to educational hypermedia: a field study, `http://www.mywire.com/a/JournalofEducationalMultimedia/Applying-Technology-Acceptance-Model-TAM/1328179/`
32. Dima, A.M., Stancov, V.: Taxonomies of Organizational Knowledge. Revista Informatica Economică 4(48), 74–76 (2008)
33. Ling, L.H.: From Shannon-Weaver to Boisot: A Review on the Research of Knowledge Transfer Model. In: International Conference on WiCom, pp. 5439–5442. IEEE Press, Los Alamitos (2007)
34. Wallin, C., Crnkovic, I.: Three Aspects of Successful Software Development Projects When are projects canceled, and why? In: Proceedings of the 29th EUROMICRO Conference, pp. 368–374. IEEE Press, Los Alamitos (2003)

Automating the Dissemination of Information Entities to Healthcare Professionals

Juha Puustjärvi[1] and Leena Puustjärvi[2]

[1] Helsinki University of Technology, Box 9210, 02015 TKK, Finland
juha.puustjarvi@tkk.fi
[2] The Pharmacy of Kaivopuisto, Neitsytpolku 10, Helsinki 00140, Finland
leena.puustjarvi@kolumbus.fi

Abstract. Lifelong learning is a term that is widely used in a variety of context. The term recognizes that learning is not confined to classroom, but takes place throughout life and in a range of situations. A problem in lifelong learning is the dissemination of relevant learning and informal material. Ideally, all the employees should receive all the relevant material while not burden with irrelevant material. Neither the searching of the material should not cause extra efforts. In trying to achieve these goals we have developed an Information entity ontology, which captures learning material and the tasks of employees' daily duties as well as their relationships. This ontology allows (i) the integration of learning material with daily duties, and (ii) the delivery of the material to relevant personnel. This kind of dissemination, however, requires that the material is augmented by extra semantic information, which is used in positioning the material into the Information entity ontology. Further in order to avoid the efforts required for searching learning material, a workflow engine is required for coordinating daily duties and attaching learning material to daily duties. In this paper, we consider the deployment of such solutions in healthcare sector.

Keywords: Learning object ontologies, Lifelong learning, Dissemination of information, Knowledge management, Semantic web, Health information systems.

1 Introduction

In healthcare sector the fast development of drug treatment and the introduction of new technologies require specialized skills and knowledge that need to be renewed frequently. This in turn entails a huge amount of educational and informal learning material that has to be disseminated for the healthcare organizations and further to their employees [1].

Most healthcare organizations receive educational and informal material from a variety of sources [2], e.g., from medical authority, medicinal wholesalers, and pharmaceutical companies. These information entities arrive in variety formats, e.g., by paper mail, e-mail, and fax. Also the nature of the information entities may vary, e.g., an information entity may be a learning object, a regulation, a guide or a bulletin.

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 123–132, 2009.

A problem here is how the dissemination of information entities should be organized. In an ideal case each employee would receive all the relevant information entities and no irrelevant information entities. In addition, the dissemination should be done in a way that the searching of the information entities would not require extra efforts.

Nowadays, the common practice in information dissemination is that the incoming information entities are stored in a variety of systems such as in Document Management systems, Learning Content Management Systems, Content Management Systems, Databases Systems and Customer Relationship Management Systems [3]. However, the problem here is that same information may be stored in separate systems, and each system is hardcoded to only work with its own data. Such hardcoded systems are problematic from the point of our goals.

To illustrate this, assume that the regulation titled “New warnings of using pain drugs with children” is received in a pharmacy from a medicinal authority. In the pharmacy, it is stored in document management system under the folder Regulation. From time to time each pharmacist check, or at least should check, the folder to see whether there are any new regulations. Assume now that a pharmacist opens the document management system and finds the regulation. After reading this warning, the pharmacist wants to know the medicinal products that the warning concerns, i.e., which medicinal products are under the classification “pain drugs for children”. To find such information the pharmacist has to open a medicinal information system and make specific searches to find such information. However, there is no guarantee that the pharmacist will succeed in finding the relevant medicinal products.

An alternative way is to store all the information entities in one data store, such that all the systems can use the same data store, and thus avoid the use of many applications within a user task. The data store may also capture information about the use of the information entities in daily routines, and so enable the automatic integration of information entities to daily routines. This is our chosen way that we will report in this paper.

First, in Section 2, we consider the nature and expression power of learning object metadata standards. We show that the expression power of the metadata standards is not enough for integrating learning objects with daily duties but rather the introduction of an appropriate ontology is needed. Then, in Section 3, we present our developed Information entity ontology, and in Section 4, we consider the sharing of the ontology among organization’s applications. In Section 5, we present how the ontology can be exploited by a workflow engine, which attaches information entities to the relevant tasks of the daily routines. Finally, Section 6 concludes the paper by discussing the advantages and disadvantages of our approach.

2 Metadata and Ontologies

2.1 Learning Object Metadata

During the past years the term learning object is widely used in the discussion concerning educational information systems. Institute of Electrical and Electronics Engineers (IEEE) defines a learning object as "any entity, digital or non-digital, that may be used for learning, education or training" [4]. Generally the term is understood to be a digital entity deliverable over Internet such that any number of learners can use them simultaneously [5].

There are four commonly accepted functional requirements set on learning objects [6]. First, learning objects should be usable in different instructional contexts, i.e., learning objects should be reusable. Second, learning objects should be independent of the delivery media and learning management system, i.e., learning objects should enable the interoperability of learning management systems. Third, learning object should be designed in the way which allows the combination of learning objects. Fourth, learning objects should provide appropriate metadata in order to allow easy searching facilities.

In order to satisfy the fourth requirement, learning objects standards like LOM [4] associate with learning material metadata such as educational and pedagogical properties, access rights and condition of use as well as the relationship to other educational resources [7].

Although these metadata [8] items are useful they do not have enough expression power to associate educational resources to other non-educational resources such as daily work patterns, processes, workflows, roles or other information entities [9]. Hence, in order to capture such concepts we have to deploy an ontology specification language, and specify an ontology that captures the appropriate concepts and the relationships between these concepts. Such things we have captured in our developed Information entity ontology.

2.2 Information Entity Ontology

The term ontology originates from philosophy where it is used as the name of the study of the nature of existence [10]. In the context of computer science, the commonly used definition is “An ontology is an explicit and formal specification of a conceptualization” [11]. So it is a general vocabulary of a certain domain. It also tries to characterize that meaning in terms of concepts and their relationships.

Essentially the used ontology must be shared and consensual terminology as it is used for information sharing and exchange. On the other hand, ontology tries to capture the meaning of a particular subject domain that corresponds to what a human being knows about that domain. It is typically represented as classes, properties attributes and values. So it consists of a finite set of concepts and the relationship between the concepts.

Figure 1 represents a simplified version of the Information entity ontology in a graphical form, where ellipses represent classes and boxes represent properties.

In this ontology the class Information entity has four subclasses: learning object, recommendation, announcement and regulation. The common denominator for these subclasses is that an employee should be familiarized with their content; from employee’s point of view the type of an information entity is inessential.

Since the intention of the ontology is to enable the automatic dissemination of information entities, we have captured two aspects of their assumed use in the ontology: (i) who needs the information entity, and (ii) where (in which tasks) the information entity is used. The former aspect is presented by the property delivery, and the latter aspect by the property relatesTo. As we will illustrate in Section 4, the workflow

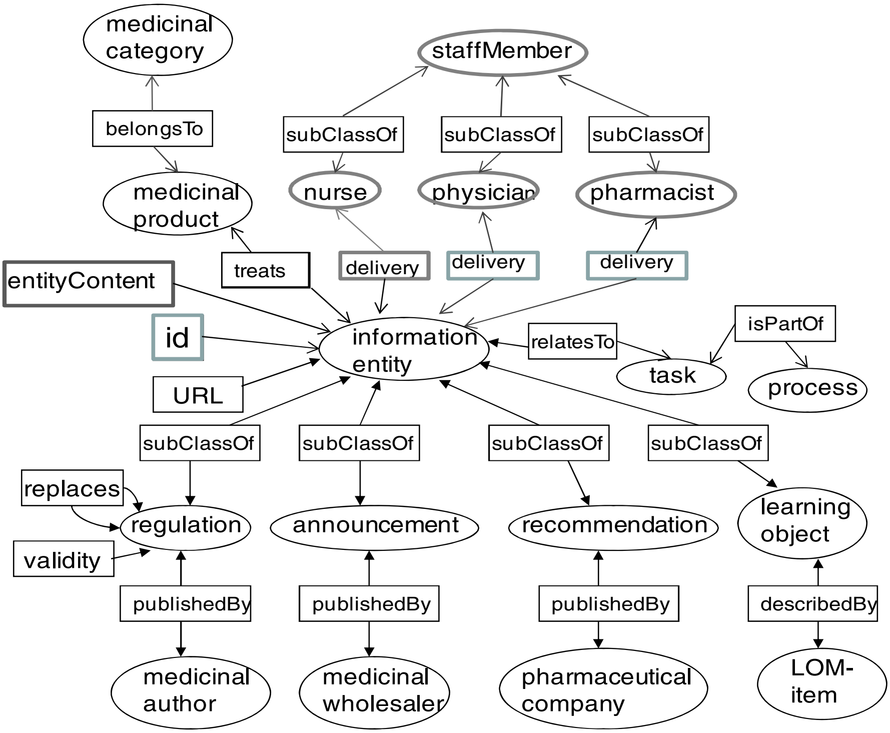


Fig. 1. Information entity ontology

engine exploits the relatesTo property in attaching information entities to the tasks of the daily routines.

The idea behind the delivery property is that it is possible attach the roles that should be aware of the information entity in question. For example assume that the delivery property has value nurse, and Lisa Fords is a nurse, then the information entity will be delivered to her.

Note that the Information entity ontology allows two ways for delivering the actual content of the entity: by giving the url of the location of the information entity, and by storing the information entity for the property entityContent.

Which way to use should be solved on a case-by-case basis, and depends on the publisher of the information entity. On the other hand, we assume that in many cases it is reasonable to use both ways. For example, the information entity may be stored in sender's own site (e.g., in medicinal wholesaler's site) in HTML whereas property entityContent may have it in printable MS Word format.

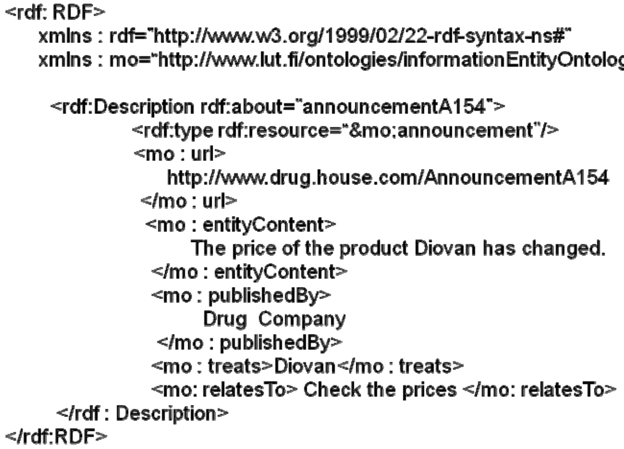
2.3 Representing Information Entities

In health care organizations the volume of incoming information entities is increasing all the time. A problem here is how we can automatically update the Information entity ontology.

We have solved this problem by representing information entities by RDF-descriptions. An RDF-description [12] specifies how the information entity relates to the Information entity ontology, which in turn is presented in OWL. That is, OWL is an ontology language that provides the vocabulary for the RDF-descriptions [13].

The RDF model is called a triple because it has three parts: subject, predicate and object. Each triple is an RDF-statement. The statements concerning the same resource is captured in an RDF-description. In order that RDF-statements can be represented and transmitted RDF needs syntax. The syntax has been given in XML. So an RDF-description can be represented as an XML-document.

In order to illustrate the presentation formats of RDF-descriptions, let us consider the XML-document of Figure 2. It is comprised of one RDF-description, which includes six RDF-statements.



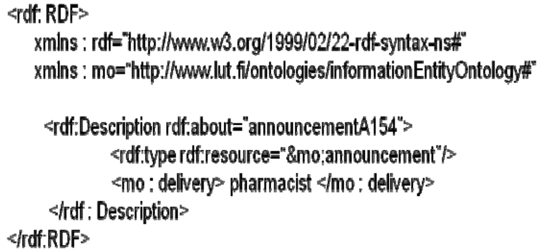
```
<rdf: RDF>
    xmlns : rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns : mo="http://www.lut.fi/ontologies/informationEntityOntology#"

     <rdf:Description rdf:about="announcementA154">
              <rdf:type rdf:resource="&mo;announcement"/>
              <mo : url>
                  http://www.drug.house.com/AnnouncementA154
               </mo : url>
                <mo : entityContent>
                     The price of the product Diovan has changed.
                 </mo : entityContent>
                 <mo : publishedBy>
                       Drug  Company
                  </mo : publishedBy>
                 <mo : treats>Diovan</mo : treats>
                 <mo: relatesTo> Check the prices </mo: relatesTo>
      </rdf : Description>
</rdf:RDF>
```

Fig. 2. An instance of an information entity presented by an RDF-description

The xml namespace mo refers to the Information Entity Ontology, which is stored at //www.lut.ontologies/InformationEntityOntology#. The type of the delivered information entity is announcement (a subclass of information entity). The id of the announcement is announcementA154. The content of the announcement is “The price of the product Diovan has changed”. This content is also stored at //www.drug.house.com/AnnouncementA154. The announcement is published by Drug Company. It treats medicinal product Diovan, and it relates to the task “Check the prices”.

Note that according to the Information entity ontology of Figure 1 we could also specify (by the property delivery) the occupational groups (nurse, physician, pharmacist) to whom the information entity should be delivered. That is, it is not necessary to give values for all the properties presented in the ontology. It is also allowed to augment the descriptions later on. For example, the RDF-description of Figure 3 states that the announcementA154 should be delivered to the occupational group called pharmacist. That is, we augmented the description concerning announcementA154, which was presented in Figure 2.



```
<rdf: RDF>
   xmlns : rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns : mo="http://www.lut.fi/ontologies/informationEntityOntology#"

    <rdf:Description rdf:about="announcementA154">
           <rdf:type rdf:resource="&mo;announcement"/>
           <mo : delivery> pharmacist </mo : delivery>
    </rdf : Description>
</rdf:RDF>
```

Fig. 3. Augmenting an RDF-description

3 Sharing the Information Entity Ontology

The idea behind knowledge centric organization is to revolve all applications around the shared ontologies [14]. It provides the means for the retrieval of knowledge for various applications such as CRM (Customer Relationship Management) system, SCM (Supply Chain Management) system and ERP (Enterprise Resource Planning) system [14] (Figure 1).

We also follow this knowledge centric strategy, which means the Information entity ontology can be accessed by a variety of applications such as Content Management systems (CMSs), Learning content management system (LCMS) and workflow engine.

A CMS is a computer application used to create, edit, manage, search and publish various kinds of digital media such as news articles, operators' manuals, technical manuals, sales guides, and marketing brochures. From our point of view they are all information entities that employees use in daily routines, and thereby should be stored in the Information entity ontology.

LCMS is a software application, or set of applications, that manages the creation, storage, use, and reuse of learning content [15]. They often store content in granular forms such as learning objects, and so they are also stored in the Information entity ontology.

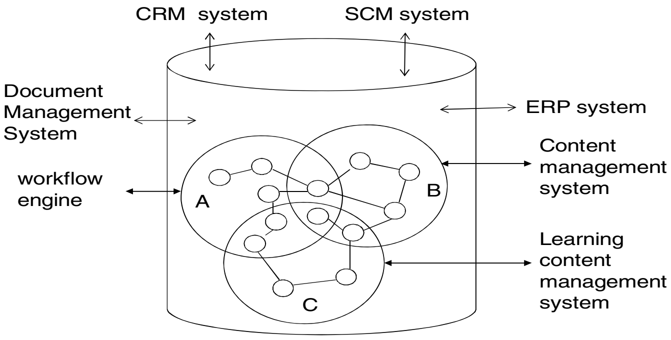

CRM system
SCM system
Document Management System
ERP system
workflow engine
A
B
Content management system
C
Learning content management system

Fig. 4. Sharing the Information Entity Ontology

The sharing of the Information entity ontology is illustrated in Figure 4. In the figure, the information entities (small circles) inside circle A are accessed by the Workflow engine, the information entities inside circle B are accessed by the CMS, and the information entities inside circle C are accessed by the LCMS, i.e., circle C represents a learning object ontology. The edges between the information entities illustrate the associations between them, e.g., as illustrated in Figure 1, there is an association relatesTo between information entity and task. Conceptually the Information entity ontology represents the union of the circles A, B and C, i.e., those entities that are accessed by CMS, workflow engine or LCMS.

4 Integrating Information Entities into Daily Work Patterns

4.1 Workflow Engine

The concept of workflow was introduced to facilitate business process automation. A workflow specification describes business tasks and their execution dependencies. Based on a workflow specification a workflow engine is able to run workflows, i.e., to coordinate the execution of the tasks of the workflow [16].

A task may be automated, partially automated or manual. Automated tasks are fully executed by the computer, e.g., the executing a program that compute the price of the product. Partially automated tasks are executed by humans by using a computer while manually tasks are wholly processed by humans.

Our idea is to extend the functionality of the workflow engine by providing appropriate guidance for the execution of partially automated and manual tasks. That is, if in the Information entity ontology there is a regulation, announcement, recommendation or learning object associated to the task, then the workflow engine informs (by providing a link to the entity) the human being about that information entity.

4.2 Modeling Workflows by BPMN

Though the ultimate goal of using Business Process Modeling Notation (BPMN) [17] is the automation of the coordination of business processes, we use BPMN to model medicinal processes and for attaching medicinal instructions to day-to-day work patterns, which are presented in BPMN.

The BPMN defines a Business Process Diagram (BPD), which is based on a flowcharting technique tailored for creating graphical models of business process operations [18]. These elements enable the easy development of simple diagrams that will look familiar to most analysts. In addition BPMN allows an easy way to connect documents and other artifacts to flow objects, and so narrows the gap between process models and conceptual models. Also, a notable gain of BPMN specification is that it can be used for generating executable workflow specification, which is presented by BPEL (Business Process Execution Language) [19] code. That is, the workflow engine coordinates the execution of the tasks according to the BPEL code. From technology point of view BPEL code is an XML-based workflow specification language.

In BPD there are three Flow Objects: Event, Activity and Gateway. An Event is represented by a circle and it represents something that happens during the business process, and usually has a cause or impact. An Activity is represented by a rounded corner rectangle and it is a generic term for a task that is performed in companies. The types of tasks are Task and Sub-Process. So, activities can be presented as hierarchical structures. A Gateway is represented by a diamond shape, and it is used for controlling the divergence and convergence of sequence flow.

In BPD there are also three kind of connecting objects: Sequence Flow, Message Flow and Association. A Sequence Flow is represented by a solid line with a solid arrowhead. A Message Flow is represented by a dashed line with an open arrowhead and it is used to show the flow of messages between two separate process participants. An Association is represented by a dotted line with a line arrowhead, and it used to associate data and text with flow objects.

4.3 A BPD Example

In Figure 5, we have presented how the process of producing electronic prescription can be represented by a BPD, and how we can attach information entities to the tasks of the diagram.

As illustrated in the figure, we use Association to attach information entities (i.e., regulations, announcements, recommendations, and learning objects) to Activities and Gateways. For example, RecommendationA7 and LearningobjectR5 are associated to activity "Produce prescription", and AnnouncementK7 is associated to gateway "Check negative effects". However, it should be emphasized that the associations are not determined by the business process designer but the workflow engine, which coordinate the execution of the process.

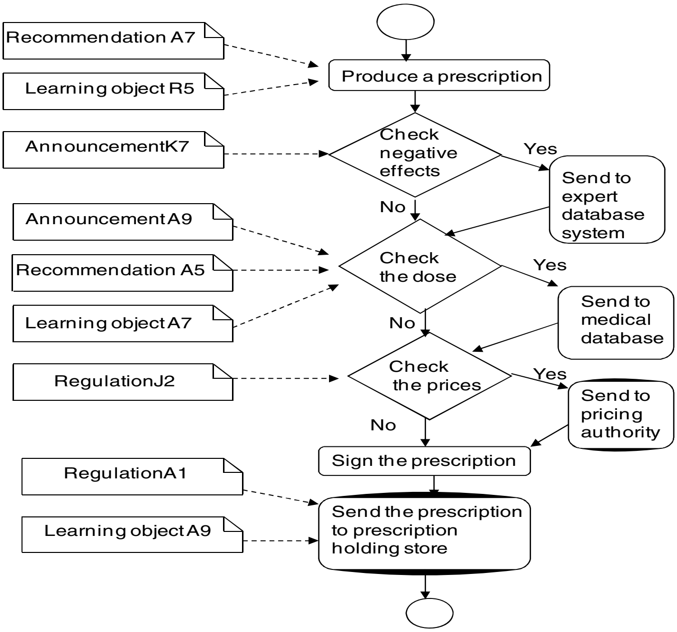


Fig. 5. Representing a medicinal process by a BPD

In attaching information entities to a task the workflow engine first has to query the information entity ontology which information entities are associated to the task. Then, after it has received the reply, it makes the association. For example, in coordinating the task "Check the dose" the workflow engine submits a query "Give the URLs of the information entities that are related to task "Check the dose". Then according to the reply the workflow engine provides the links to the user. The user may open the entities by clicking the URLs. Hence the actual location of an information entity may be in any organization. Further, the entity may be in a human readable form such as in HTML [14] or in machine readable form such as in XML [14], which is then transformed by a style sheet engine into HTML.

5 Conclusions

Lifelong learning is a term that is widely used in a variety of context. The term recognizes that learning is not confined to classroom, but takes place throughout life and in a range of situations.

We have analyzed lifelong learning in the context of workers daily duties in healthcare sector, where the fast development of drug treatment requires special knowledge. Our ultimate goal has been the integration of learning processes and daily duties in a way that searching educational material does not require extra efforts.

The advantage of attaching learning objects as well as other relevant information into daily routines is that we can ensure that the workers will not loss relevant information as a result of huge amount of incoming information.

From technology point of view our approach is based a knowledge base and a workflow engine, which coordinates the execution of the daily duties. The introduction of these technologies also changes the daily duties of the many employees of the healthcare organization. Therefore the only challenging aspect is not the technology but also changing the mind-set of the employees and the training of the new technology.

References

1. Batenburg, R., Van den Broek: Pharmacy information systems: the experience and user satisfaction within a chain of Dutch pharmacies. International Journal of Electronic Healthcare 4(2), 119–131 (2008)
2. Khoumbati, K., Shah, S., Dwivedi, Y.K., Shah, M.H.: Evaluation of investment for enterprise application integration technology in healthcare organisations: a cost-benefit approach. International Journal of Electronic Healthcare 3(4), 453–467 (2007)
3. Puustjärvi, J., Puustjärvi, L.: Managing personalized and adapted medical learning objects. In: The Proc. of the 7th IEEE International Conference on Advanced Learning Technologies, ICALT (2007)
4. LOM – Learning Object Metadata, `http://ltsc.ieee.org/wg12/`
5. Puustjärvi, J.: The role of metadata in e-learning systems. In: Ma, Z. (ed.) The book "Web Based Intelligent e-Learning Systems: Technologies and Applications". Idea Group Inc., USA (2005)

6. Puustjärvi, J.: Syntax and Semantics of Learning Object Metadata. In: Harman, K., Koohang, A. (eds.) The book “Learning Objects: Standards, Metadata, Repositories, and LCMS”. Informing Science Press (2007)
7. Hatzilygeroudis, I., Prenzas, J.: Knowledge Representation in Intelligent Educational Systems, Web-Based Intelligent E-Learning Systems. In: Ma, Z. (ed.) The book “Web Based Intelligent e-Learning Systems: Technologies and Applications”. Idea Group Inc., USA (2005)
8. Puustjärvi, J.: The role of metadata in e-learning systems. In: Ma, Z. (ed.) The book “Web Based Intelligent e-Learning Systems: Technologies and Applications. Idea Group Inc., USA (2005)
9. Puustjärvi, J.: Syntax and Semantics of Learning Object Metadata. In: Harman, K., Koohang, A. (eds.) The book “Learning Objects: Standards, Metadata, Repositories, and LCMS. Informing Science Press (2007)
10. Gruber, T.R.: Toward principles for the design of ontologies used for knowledge sharing. In: Padua workshop on Formal Ontology (March 1993)
11. Antoniou, G., Harmelen, F.: A semantic web primer. MIT Press, Cambridge (2004)
12. RDF – Resource Description Language, `http://www.w3.org/RDF/`
13. OWL – WEB OntologyLanguage, `http://www.w3.org/TR/owl-features/`
14. Daconta, M., Obrst, L., Smith, K.: The semantic web.:A Guide to the Future of XML, Web Services, and Knowledge Management. John Wiley & Sons, Chichester (2003)
15. Irlbeck, S., Mowat, J.: Learning Content Management Systems (LCMS). In: Harman, K., Koohang, A. (eds.) The book “Learning Objects: Standards, Metadata, Repositories, and LCMS. Informing Science Press (2007)
16. Dwivedi, A., Bali, R.K., Wickramasinghe, N., Naguib, R.: How workflow management systems enable the achievement of value driven healthcare delivery. International Journal of Electronic Healthcare 3(3), 382–393 (2007)
17. BPMN - Business Process Modeling Notation (BPMN) Information, `http://www.bpmn.org/`
18. White, A.: Introduction to BPMN, IBM Corporation, `http://www.bpmn.org/Documents/Introduction%20to%20BPMN.pdf`
19. BPEL4WS – Business Process Language for Web Sevices, `http://www.w.ibm.com/developersworks/webservices/library/ws-bpel/`

The Triangular Life Cycle Model

Phil Robinson

Lonsdale Systems
Perth, Australia
lonsdale@iinet.net.au

Abstract. The waterfall life cycle model suffers from a number of problems but in spite of this, it continues to be the most widely used life cycle model. A different life cycle approach is proposed that emphasizes the product life cycle rather than the project life cycle, quality management priorities rather than project management priorities and views of quality rather than views of the project schedule. A quality management tool based on different views of quality is used to identify the "gaps" that inevitably exist between a user's needs, the requirements specification and the product that is delivered. This is followed by a brief discussion of how these "gaps" can be closed.

Keywords: Life Cycle, Model, Quality, Software, Waterfall.

1 The Waterfall Life Cycle Model

The waterfall life cycle model is the best known and most widely used life cycle model. A recent survey found that more than a third of organisations still base their software development projects on this life cycle model [1].

The introduction of the waterfall life cycle model is frequently attributed to Winston Royce [2]. Interestingly, the phrase "waterfall life cycle model" is not mentioned in Royce's article. In fact, the article appears to argue for a more iterative approach to software development!

If Royce is not the source of the waterfall life cycle model, it is possible that the phrase was first used as a metaphor for the sequential phases of a project.

PMBOK – the definitive guide to project management best practice [3] – describes a project life cycle model that groups project activities into a number of sequential phases. According to PMBOK, the project life cycle "defines the phases that connect the beginning of a project to its end", a project phase is characterized by "the completion and approval of one or more deliverables" and "phases are generally sequential and are usually defined by some form of technical information transfer or technical component handoff".

In software projects, the "sequential" organization of project phases is usually interpreted as the need to group project activities based on the deliverable they create. This leads to the familiar project phases of requirements, design, coding and testing.

PMBOK phases include an initial phase, a number of intermediate phases and a final phase. Again, according to PMBOK, the final phase "includes the processes used to formally terminate all activities of a project" and "hand off the completed product to others"

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 133–146, 2009.

For many projects, a life cycle that comes to a conclusive end makes sense. It is true; that once a building project is finished there is little more to do other than move the new tenants into the building. If the project has remained on schedule and within the budget, the project manager will most likely receive well-deserved praise. The project team will either be disbanded or move on to the next project.

In contrast, many software products play a crucial role in sustaining a business either as products in their own right or by supporting business processes. This means that the "final phase" of a software project is only reached when a product is eventually retired.

2 The Triangular Life Cycle Model

Over the years, many refinements to the waterfall model have been suggested and alternative life cycle models proposed.

While there is always a lot of interest in improving on the waterfall model, many organisations are found lacking when it comes to actually implementing improvements [1]. One of the reasons behind the resistance to change could be that many of the alternative approaches are based on elaborate concepts and sometimes accompanied by all-embracing ideologies [4]. This can make them difficult for more pragmatic, outcomes-focused project managers to accept.

With this in mind, the Triangular Life Cycle Model (TLCM) is based on the simple concept of a triangle combined with three fundamental principles that emphasize:

- the product life cycle rather than the project life cycle;
- quality management priorities rather than project management priorities; and
- views of quality rather than views of the project schedule.

2.1 Product Life Cycle

The product life cycle commences with the needs, wants and expectations of its users. These are captured as the product requirements on which the development of the product is based.

Once it is put into operation, the users will most likely identify opportunities for the product's enhancement and refinement. These opportunities lead to a revised set of user needs, wants and expectations. These in turn lead to a new set of requirements and ultimately a new version of the product.

For some software products, this cycle of on-going change and refinement may go on for many years. In some cases the same core team is responsible for development of the product over this time.

Frustrated by the inconsistencies between the project and product life cycles, some software developers have turned to the Japanese concept of "wabi-sabi" in search of a better model for software development. Wabi-sabi is an aesthetic principle that based on the acceptance of transience – "nothing lasts, nothing is finished, nothing is perfect" [5].

Another important area of difference between the product and project life cycles is the measure of success. For the product life cycle, success is measured by how well the final product meets it user's needs, in other words by quality and scope. In contrast, success for the project life cycle usually emphasizes time and cost.

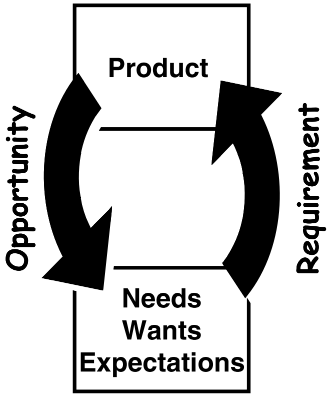


Fig. 1. The product life cycle

2.2 Quality Management Priorities

PMBOK identifies nine areas of knowledge required by a project manager. The knowledge areas include topics such as the management of human resources, communication and risk. However, for most project managers the four most important areas and their order of priority are time, cost, scope and quality.

One of the effects of emphasizing quality management rather than project management is to invert the order of priorities to quality, scope, cost, time.

The reason for quality's place at the top of the list is self-evident. Scope is the second item because a product's ability to satisfy its user's needs is a fundamental measure of quality and also the product life cycle measure of success. Cost appears before time because the cost of quality is a well-defined concept [6] that measures both the cost of poor quality and the cost of achieving good quality.

However the placement of time at the bottom of the list does not mean that it is unimportant but rather that from a quality perspective, it is less important than the other priorities.

2.3 Views of Quality

It is not surprising that Gantt charts are the universal tool for planning and monitoring projects. Gantt chats use horizontal bars to represent time, which are usually a project manager's top priority. To emphasize the position of quality as the first priority in the TLCM, a similar tool is required. David Garvin's views of quality [7] provide a good starting point for developing such a tool. Garvin identifies five views of quality:

- Transcendental – this view of quality associates quality with "innate excellence" that is "absolute and universally recognizable". This view is useful for marketing products or establishing brands but because of its subjective nature, not so useful for quality improvement.
- User – this view of quality focuses on the ability of a product to satisfy the needs of its users.
- Manufacturer – this view associates quality with "conformance to (engineering and manufacturing) requirements". It focuses on how well a product conforms to its specification.

- Product – this view of quality associates quality with product characteristics that can be measured using “a precise and measurable variable”. It focuses on measurable attributes of a product.
- Value – this view of quality measures quality “in terms of costs and prices”. A quality product is one that provides performance at an acceptable price or conformance at an acceptable cost.

Three of these views have been selected as the basis for the quality tool:

- the user’s view which is represented by the user’s needs;
- the manufacturer’s (software developer) view which is represented by the requirements specification; and
- the product view.

The value view of quality is implied by the user’s needs, which include the price they are prepared to pay for the product and the manufacturers view, which includes the cost of developing the product. The transcendental view of quality is probably best left to those who market and develop product brands.

Except in the case of an imaginary “perfect” product, it is unlikely that the user’s needs, the specification and the final product will all be in perfect alignment. This will lead to discrepancies or “gaps” between the user, manufacturer and product views of quality.

The gaps between the three views of quality can be represented by a triangle with one of the views placed at each corner of the triangle.

- The need-specification gap represents how well the specification describes the user's needs.
- The specification-product gap represents how well the product conforms to its specification.
- The product-need gap represents how well the final product satisfies the user's needs.

The length of the sides represents the magnitude of the gap between the views. Since the length of any side of a triangle always depends on the length of the other two sides, the magnitude of the product-need gap experienced by the users of the product, will always depend on the magnitude of the need-specification and specification-product gaps.

In other words, there are two different scenarios that can result in a product ultimately not meeting the needs of its user:

- a poor understanding of the user’s needs which results in a need-specification gap; or
- not following the specification, which results in a specification-product gap.

Six Sigma is a business improvement strategy that identifies the same two scenarios as sources of dissatisfied users referring to them as “the voice of the customer” and the “voice of the process” [8].

Users of software products often have difficulty articulating their needs and providing feedback on requirements specifications describing the developer’s understanding of their needs. This often leads to numerous changes to the software product completed when they see it for the first time they realize that it is not what they require.

Barry Boehm has described this as the, “I’ll Know It When I See It” (IKIWISI) phenomenon [9].

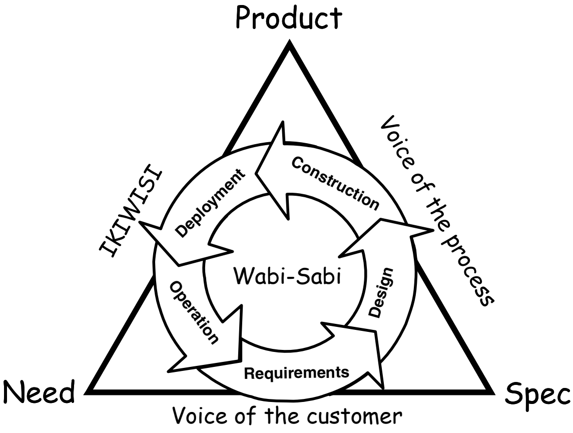


Fig. 2. The Triangular Life Cycle Model

Although time appears as the last priority in the TLCM, it is obviously still an important aspect of software projects. Time is added to the quality triangle by superimposing a number of sequential life cycle stages onto the triangle. To align properly with the views of quality and the gaps between them, the life cycle stages are arranged into a circle – this also reflects the cyclic nature of the product life cycle.

The requirements stage of the life cycle contribute to the need-specification gap, while design and construction stages contribute to the specification-product gap. The magnitude of the product-need gap is determined during the deployment stage and is experienced by the users during the operation stage.

2.4 The Role of Verification and Validation

Project teams are often confused about the different objectives of "verification" and "validation". The TLCM provides an opportunity for some clarification. Validation is

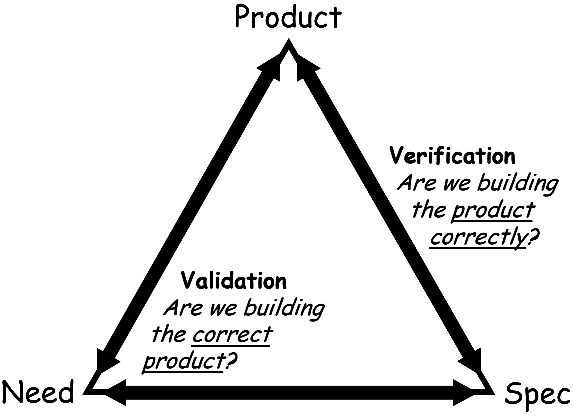

Product
Verification
Are we building the product correctly?
Validation
Are we building the correct product?
Need
Spec

Fig. 3. Verification and validation

represented on the triangle in two places: between the user's need and the specification; and between the product and the user's need.

In both cases, validation answers the question, "are we building/have we built the correct product"?

Verification is shown on the remaining side of the triangle between the specification and the product. It answers the question, "are we building the product correctly"

3 Closing the Gaps

Verification and validation (V&V) are the classic techniques used to close the gaps between the different views of quality. In addition to V&V there a number of other techniques that can be used – configuration management, defect prevention, rework, iteration and process improvement.

3.1 Closing the Gaps with Verification

Verification is a technique for closing the specification-product gap during the design and construction stages of the life cycle. It achieves this by identifying discrepancies between the product and the specification. The discrepancies can then be corrected before construction of the product is completed.

As well as the final product, there are many interim work products that need to be developed during the life cycle. Many of these work products are documents such as architectural designs, detailed designs or test plans. Interim work products such as these can be verified against the work products from which they are derived. For example, a test plan could be verified against a detailed design document, an architectural design document as well as the requirements specification.

Testing is one the techniques that is most frequently used for verification. Testing is defined as "the process of exercising software to verify that it satisfies specified requirements; and to detect errors" [10].

Traditionally there are three different ways of testing a software product during construction phase of the life cycle – component testing, integration testing and system testing.

However, there are numerous work products that cannot be tested because they cannot be "exercised" (executed). For example, it is not possible to exercise documents, models or source code.

Reviews are a means of verifying work products that cannot be exercised. The IEEE standard for software reviews [11] describes four types of review that can be used for verification – technical reviews, inspections, walk-throughs and audits.

3.2 Closing the Gaps with Validation

Validation appears twice in the TLCM. Requirements validation takes place during the requirements stage of the life cycle and is a technique for closing the need-specification gap. It achieves this by ensuring that the specification accurately describes the user's needs, wants and expectations.

Product validation takes place during the deployment and operation stages of the life cycle but it can only be used to measure the magnitude of the product-need gap.

At these late stages of the life cycle, backtracking and rework will be required to actually close the gap. Product validation determines how well the completed product satisfies the user's needs.

3.2.1 Requirements Validation

There are many different techniques that can be used for requirements validation. Four of the most popular techniques are – workshops, modelling, prototypes and user requirement reviews.

Workshops are good technique for ensuring user participation and resolving conflicting requirements. Workshops must have clear objectives and will require an experienced workshop facilitator who is responsible for ensuring that the workshop achieves its objectives.

Natural language is inherently ambiguous. This makes it a poor choice for the precise description of requirements. In contrast, diagrams and models have the ability to describe requirements with less ambiguity. Diagrams and models are often more compact, easier to change and better at enforcing consistency than natural language. Modelling standards such as the UML [12] have further enhanced the clarity of diagrams and models.

Prototypes are a way to address the IKIWIS phenomenon by allowing users to "see" the product early in its life cycle. A prototype is a working model of the final product that can be demonstrated to (or possibly used by) users. User feedback on the prototype can be incorporated into the final specification.

User requirements reviews are a type of technical review that includes participation by the users. They provide an opportunity for the users to provide feedback on the specification and ultimately confirm that it will serve as a reasonable basis for the development of the product.

3.2.2 Product Validation

Testing is the most common technique used for product validation. While there can be many types of validation testing, acceptance testing is the type most commonly encountered.

Because acceptance testing can only measure the magnitude of the product-need gap, it is better viewed as an important life cycle milestone rather than as a technique for closing the gaps of the quality triangle.

It is a widely held belief that reviews are inherently a verification technique. However, this is not the case. For example, a user walk-through is often used as a technique for validating a simple enhancement to a product.

Another use of reviews as a validation technique is to conduct a post implementation review after a product has been in operation for some time. A post implementation review validates the product in its operational environment and ensures that the product continues to meet the user's needs.

3.3 Testing as a V&V Technique

Arranging the different types of testing around the quality triangle provides some insight into the somewhat limited role of testing as a V&V technique. As can be seen, the "testing region" encompasses only a relatively small area of the triangle and thus has a limited role in closing the gaps.

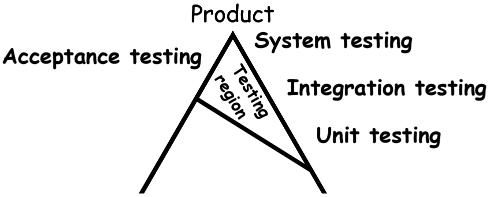


Fig. 4. The testing region

3.4 Closing the Gaps with Configuration Management

Configuration management is concerned with the correct assembly of a product from its component parts. It is a management practice designed to ensure that the correct version of a component is used for each "build" of the product and that changes to the product and its components can be controlled, traced and tracked over time. [13].

Configuration management can be used as a technique to close the specification-product gap during the design and construction phases of the life cycle. It achieves this by formally identifying different versions of a product and its components and by controlling changes to the product, its specification, its components and other interim work products.

A product may be assembled incorrectly as a result of selecting the wrong components or the wrong version of a component. Different versions of a product and its components will exist at different points in time. In addition, variants of a product may be created to meet the needs of different users and operational environments.

A product that is assembled from the incorrect components is unlikely to conform to its specification. This effectively leads to an increase in the magnitude of the specification-product gap. Positive identification of components coupled with version control helps to ensure the correct assembly of a product and will close the gap between the specification and the product.

Change can have a subtle effect on the magnitude of the specification-product gap. Changing user needs after development has commenced will increase the magnitude of the specification-product gap but the increase will not be reflected in the specification. This means that the developers are often not aware of the increased gap.

The increased gap is often not discovered until acceptance testing performed during the deployment stage. The solution to this problem is to ensure that the understanding of user needs continues to be updated during the design and construction stages of the life cycle.

3.5 Closing the Gaps with Defect Prevention

Activities performed during the requirements, design and construction stages of the life cycle "inject" defects into a product, its specification, its components and other interim work products. The role of V&V is to identify these defects so that they can be removed.

In addition to removing individual defects, it is possible to identify entire classes of defects by performing error analysis – this involves collecting and analysing data for a large number of individual defects [14]. Bug taxonomies provide a good example of some generic classes of defect [15].

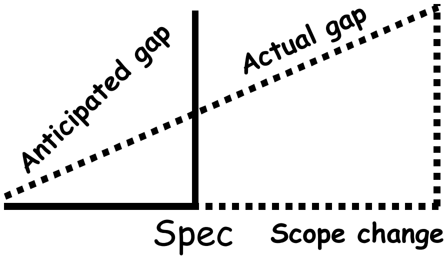


Fig. 5. The effect of change on the specification-product gap

Defect classes can be used to predict the types of defect that will most likely be injected in the future and to take some form of corrective action to prevent this from occurring. Determining the underlying or "root" cause of a class of defects often helps to identify the most appropriate corrective action. Classes of defect can also be used to improve V&V activities by providing guidance on the most likely types of defect that will be found during testing and reviews.

Defect prevention can be used as a technique to close the specification-product gap during the requirements, design and construction phases of the life cycle. It achieves this by preventing the magnitude of the gap from growing as a result of defects.

3.6 Closing the Gaps with Rework

Removing defects from a product, its specification, its components and other interim work products will normally require working backwards through life cycle to correct earlier errors and mistakes. Many activities that have already been performed will need to be performed again and many components and work products that have previously been completed will need to be modified.

The need to backtrack and revisit earlier life cycle activities is often referred to as "rework". Rework leads to additional development costs because activities are performed more than once. However, rework adds no value to the product as it simply corrects earlier errors.

Rework is an error prone activity that often injects many new defects into a product. These new defects will lead to more rework in order to remove them. The result is that rework can become a vicious circle that leads to large schedule and budget overruns.

Rework is probably the technique most widely used to close the need-specification gap during the requirements phase of the life cycle and the specification-product gap during the design and construction phases of the life cycle. This is in spite of the fact that it is the least effective technique.

3.7 Closing the Gaps with Iteration

Iteration involves performing life cycle activities more than once. Although this may sound similar to rework, iteration is quite different. Rework consists of unplanned activities required to remove defects.

Iteration on the other hand, involves the successive refinement of a product, its specification, its components or other interim work products by repeating the stages of the life cycle.

It is important that each iteration is a planned with clear objectives, outcomes and deliverables in mind [16]. Many iterative life cycles are based on the following objectives [9]:

- Definition of the "Life Cycle Objectives" (LCO) in the form of the most important requirements together with their priority.
- Definition of the "Life Cycle Architecture" (LCA) in the form of an executable architecture that will support the most important requirements.
- Delivery of an "Initial Operational Capability" (IOC) that will allow the users to perform the first acceptance test.

A software project planned around these objectives might define

- LCO during iteration 1 by delivering a prototype product.
- LCA during iteration 2 by delivering a proof of concept architecture
- IOC during iteration 3 allowing the users to acceptance test the product.

Iteration provides an opportunity for additional validation in the form of an iteration review. The findings of the iteration review serve as a major input to the planning of the next iteration.

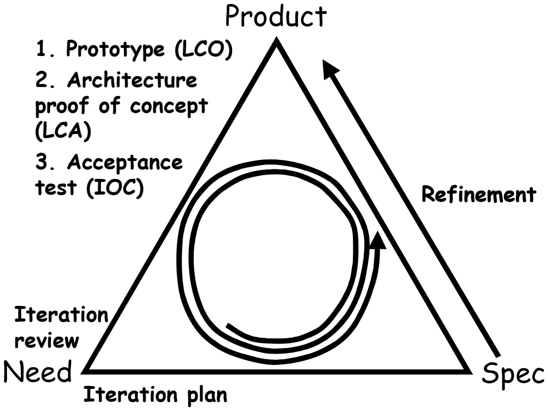


Fig. 6. Closing the gaps with iteration

Iteration can be used as a technique to close all three gaps. It achieves this by repeating the life cycle stages thus providing multiple opportunities to close the gaps.

3.8 Closing the Gaps with Process Improvement

Process improvement seeks to improve the quality of life cycle activities and their outputs. Although improving quality is normally the major focus of process improvement, it can also be applied to other aspects such as improving productivity or reducing costs.

However, improvements such as these are often only achieved as a result of improving quality. The reason for this is the manner in which quality contributes to the overall cost of a product. Quality related costs have two components:

- the cost of poor quality primarily resulting from rework but may also including the cost of product support, product updates, complaint handling, concessions to customers and loss of sales; and
- the cost of performing activities intended to close the gaps such as verification, validation, configuration management, defect prevention and additional activities associated with iteration.

The cost of poor quality is represented on the triangle by the product-need gap while the cost of closing the gaps is represented by the need-specification and specification-product gaps.

Spending money on closing the need-specification and specification-product gaps will result in a reduction in the magnitude of the product-need gap and a corresponding improvement in quality. If the increased spending on activities designed to close the gaps leads to an equal reduction in the cost of poor quality, then the improvement in quality has been achieved at no additional cost [17].

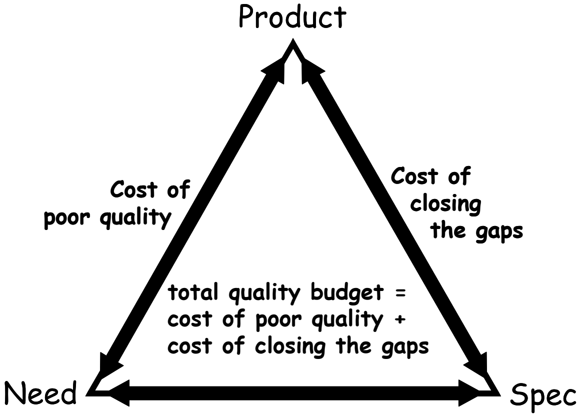

Product
Cost of poor quality
Cost of closing the gaps
total quality budget = cost of poor quality + cost of closing the gaps
Need
Spec

Fig. 7. Quality related costs

Because the improvement of software development processes normally starts from quite a poor level of quality, it is not difficult to achieve a reduction in the cost of poor quality that is greater than the amount that has been invested in closing the gaps.

Process improvement can be used as a technique to close the need-specification gap during the requirements phase of the life cycle and the specification-product gap during the design and construction phases of the life cycle.

However, there will always be a time delay between spending more on closing the gaps and a corresponding reduction in the cost of poor quality. This means that process improvement should be viewed as an investment proposition that will provide a return on the investment (ROI) at some point in the future.

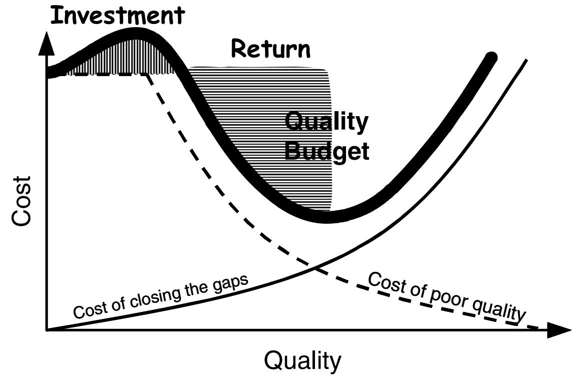


Fig. 8. Investing in process improvement

4 "Triangular" Maturity Models

The relative length of its sides determines the shape of a triangle. It is interesting to compare the shapes of triangles that reflect different project scenarios. The ideal triangle has a small need-specification gap and a small specification-product gap. The result is a product that meets most of the user's needs as represented by the small product-need gap.

The communicate triangle represents a situation in which the need-specification gap is large but is corrected by the developers during the design and construction stages of the life cycle. This is usually achieved by communicating frequently with the users - hence the name of the triangle. This triangle can deliver products that meet the user's needs but will normally involve more rework than the ideal triangle.

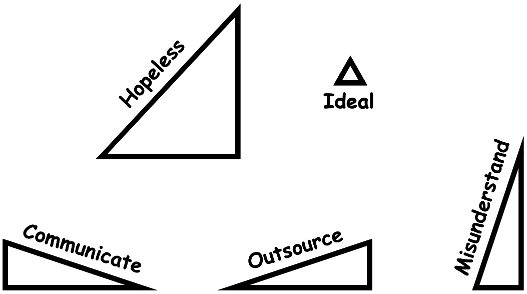


Fig. 9. Triangular maturity models

The outsource triangle has a large needs-specification gap but a small specification-product gap. The result is a product that fails to meet many of the user's needs. This triangle often occurs when the design and construction stages of the life cycle are outsourced to a third party who delivers a product that conforms closely to its specification but the specification does not reflect the true user's needs.

The misunderstand triangle has a small needs-specification gap but a large specification-product gap. The result is a product that fails to meet a many of the user's needs. This triangle can occur for two reasons:

– the specification is very complex and difficult for the developers to understand; or
– the developers do not follow the specification.

The hopeless triangle is, well simply hopeless! Large need-specification and specification-product gaps result in a product that manages to satisfy very few of the user's needs.

5 Conclusion

Project management best practice such as those described by PMBOK are intended to have relevance to a wide variety of projects undertaken in many different industries. By aiming for universal relevance these practices often miss some of the unique subtleties of software development.

The TLCM is intended to fill this gap. It is not intended as an alternative to project management practices but rather a way to supplement and enhance them with a software engineering perspective. The practices described in PMBOK can and should be applied to projects based on TLCM. However, it is hoped that the TLCM's priorities of quality, scope, cost and time can provide a useful counterweight to the more dominant time, cost, scope and quality priorities of project management.

References

1. Laplante, P.A., Neill, C.J.: The Demise of the Waterfall Model and Other Urban Myths. Game Development 1(10) (February 2004)
2. Royce, W.: Managing the Development of Large Software Systems. In: Proceedings of IEEE WESCON (1970)
3. A Guide to the Project Management Body of Knowledge (PMBOK), The Project Management Institute (also available as IEEE Std 1490-2003) (2000)
4. Beck, K.: eXtreme Programming explained. Addison-Wesley, Reading (2000)
5. Powell, R.R.: Wabi Sabi Simple, Adams Media (2004)
6. AS 2561-1982: Guide to the determination and use of quality costs, Standards Australia (1982)
7. Garviv, D.: What Does 'Product Quality' Really Mean? Sloan Management Review, 25–45 (Fall 1984)
8. George, M.L., et al.: The Lean Six Sigma Pocket Toolbook: A Quick Reference Guide to 100 Tools for Improving Quality and Speed. McGraw-Hill, New York (2004)
9. Boehm, B.: Escaping the software tar pit: model clashes and how to avoid them. In: SIGSOFT Software Engineering Notes, January 1999, vol. 24 (1999)
10. Glossary of Software Testing Terms, `http://www.testingstandards.co.uk/glossary.htm` (retrieved on August 1, 2008)
11. IEEE Std 1028-1997 IEEE Standard for Software Reviews, Institute of Electrical Engineers (1992)

12. OMG Unified Modeling Language (OMG UML), Superstructure, V2.1.2, Object management Group, OMG (2007)
13. Bersoff, E.H.: Elements of Software, Configuration Management. In: Dorfman, M., Thayer, R.H. (eds.) Software Engineering, IEEE Computer Society Press, Los Alamitos (1997)
14. Peng, W.W., Wallace, D.R.: Software Error Analysis. NIST Special Publication 500-209 (1993)
15. Beizer, B.: Software Testing Techniques. Van Nostrand Reinhold, New York (1990)
16. Boehm, B.: A spiral model of software development and enhancement. In: SIGSOFT Software Engineering (1986)
17. Cosby, P.: Quality is Free, New American Library (1979)

A Hybrid Technique for Complete Viral Infected Recovery

Pawut Satitsuksanoh, Peraphon Sophatsathit, and Chidchanok Lursinsap

Advanced Virtual and Intelligent Computing (AVIC) Center
Department of Mathematics, Faculty of Science
Chulalongkorn University
Bangkok, Thailand
numbkrub@yahoo.com, peraphon.s@pioneer.netserv.chula.ac.th,
lchidcha@pioneer.netserv.chula.ac.th

Abstract. This research proposes a hybrid technique for computer virus detection and recovery. We made use of the well-established BWT to pinpoint where the infection was located. To insure perfect detection, the CRC technique was supplemented. In the mean time, the original uninfected code was analyzed to obtain necessary unique identifications, whereby recovery process can be carried out directly with reference to these unique identifications. The proposed technique was gauged against a couple of commercial anti-virus software and found to perform its task to perfection.

Keywords: computer viruses, virus detection and disinfection, BWT compression, data integrity check, information security.

1 Introduction

Anti-virus software today is fairly sophisticated, but virus writers are often a step ahead of the software. New computer viruses are constantly being released which the current anti-virus software cannot recognize. Most anti-virus systems are still based on scanning detection using virus signature because of their very low false alarm [1, 2]. To get a new virus signature, the anti-virus researcher has to analyze the infected code of a host file in order to extract the specific pattern of a particular virus before releasing a new updated signature file. This process may take quite a long time for complicated coding viruses for instance, armored virus, polymorphic or metamorphic virus. This is a main drawback of using signature based virus detector. We are interested in not only the problem of detecting virus but also the problems of disinfecting and cleaning virus from the target program. There are many kinds of virus which destroy or replace target files. Existing commercial anti-virus systems cannot recover back the healthy program from these kinds of infection. The only possible solution is to delete the infected file and reinstall from the previously back up file. From the previously stated drawbacks, we have proposed a framework to create a file archive together with message digest for virus, change detection, file recovery, and virus cleaning.

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 147–159, 2009.

Details of the proposed technique will be described in subsequent sections. This paper is organized as follows. Section 2 describes background in computer virus. Some fundamental techniques are described in related work of Section 3. The proposed technique is elucidated in Section 4, along with experimental results in Section 5. Some final thoughts are given in Section 6.

2 Background

In this research, we focus on real computer viruses which infect or change the contents of files. These viruses can be classified by the way they operate on the host file.

2.1 Classification of Virus Infection Techniques

Computer viruses can be classified according to different aspects such as target format, behavior of each virus, payload type, etc. A popular technique is based on infection techniques [1, 2, 3] as follows:

Overwriting viruses. This infection technique simply overlays part of the existing target code with the virus own copy. The size of the infected files may increase or decrease if it is completely replaced by the virus code. The infected file may have the same size as the original one if it is partly replaced with viral code. Overwriting viruses cannot be disinfected from a system by the existing anti-virus program. Infected files must be deleted from the disk and restored from backups.

Adding viral code: appenders and prependers. The technique gets its name from the location of the virus body, which is added at the beginning or the end of the target program. This method will inevitably increase the size of the infected file unless a stealth technique is applied.

Code interlacing infection or hole cavity infection. This infection technique typically does not increase the size of the infected target. The cavity virus overwrites a portion of the file to safely store the virus code. It typically overwrites areas of files that contain zeros in binary files or code areas that have been allocated by the compiler but only very partially used by the code itself.

Companion viruses. This infection technique is quite different from all previously mentioned techniques. The target code is not modified, thus preserving the code integrity. The companion virus operate as follows. The viral code identifies a target program to attack and create an additional file, which is somehow linked to the target code to be executed in place of the target file.

2.2 Anti-virus Techniques

The most efficient modern anti-virus applications have combined several different techniques [1, 2, 3] which are briefly described below.

Searching for virus signature. This technique searches for any known sequence of bits which distinguishes a particular infected program from other programs. This technique is still used by most commercial anti-virus programs because it can detect known viruses efficiently. However, this technique fails to handle unknown or armored viruses such as polymorphic viruses or metamorphic virus. A major drawback of this technique is that it must keep the virus signature database up-to-date and secured during distribution and use.

Spectral analysis. This technique statistically analyzes instructions of a given program to find subsets of unusual instructions or contain feature specific to viruses. Thus, this technique may cause many false alerts. Fortunately, the advantage of this technique is that some unknown viruses may be detected by incorporating into other known techniques.

Heuristic analysis. This technique uses rules and strategies to study how a program behaves. The purpose is to detect potential virus activities or behavior. The advantage and drawback of this technique are similar to spectral analysis which can detect unknown viruses but produce more false alerts.

Activity monitoring. This technique monitors various activities of viral programs by being memory-resident to detect and stop any potential suspicious activities. This technique may sometimes succeed in both detecting unknown viruses and avoiding infections. The drawbacks are producing more false alert, requiring frequent update of virus behavior database, and degrading system performance as it operates in real-time mode.

Code emulation. This technique utilizes a virtual machine to mimic code execution under CPU and memory management systems. Thus, infected code is simulated in the virtual machine of the scanner having no actual virus code executed by the real processor. This technique can detect encrypted, polymorphic, and metamorphic viruses at the expense of computer resources and time.

File integrity check or change detection. This technique aims at monitoring and detecting any modification of sensitive files such as executables, documents, etc. Traditionally for each file, the file digest is computed with the help of either hash function such as MD5 or SHA-1, or cyclic redundancy codes (CRC) [4]. Our proposed technique is in this category. There is a known issue of using CRC for the purpose of virus detection or file integrity check is vulnerability to be exploited by the virus writer [4, 5]. This is not the case for our proposed technique because CRC is used as the supplementary check in the message digests.

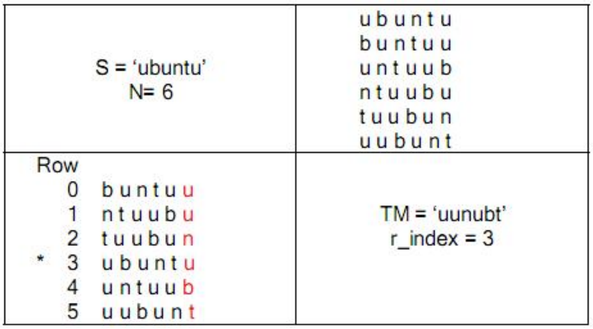
2.3 BWT Compression

Our proposed technique is primarily based on Burrows-Wheeler Transformation (BWT) [6]. BWT is the heart of a compression algorithm. The BWT itself is not a compression technique but permutates the original data to be more compressible for further processing.

The first step of BWT compression is to take a string S of N symbols S[0], S[1], ..., S[N-1] and construct the N rotation strings such that:

S[0], S[1], ..., S[N-2], S[N-1]
S[1], S[2], ..., S[N-1], S[0]
...
S[N-1], S[N-2], ...,S[1], S[0]

A table of N rows is formed and sorted lexicographically. The output of transformation is the last column and the index which is called r_index in this research. The attribute of r_index is the reverse BWT.An example of the transformation over the string, 'ubuntu' is shown in Figure 1.



S = 'ubuntu'
N= 6
	ubuntu
	buntuu
	untuub
	ntuubu
	tuubun
	uubunt

Row
0 buntuu
1 ntuubu
2 tuubun
* 3 ubuntu
4 untuub
5 uubunt
	TM = 'uunubt'
	r_index = 3

Fig. 1. An example of performing BWT over string S = 'ubuntu'

The transformed block is further processed by Move-to-Front (MTF) and Run Length Encoding (RLE) function before it is compressed by the Compression Module using entropy encoding techniques such as Huffman encoding or Arithmetic encoding. Details on how it works can be found in [6, 7, 8].

3 Related Work

Because of the limitation in detecting unknown computer viruses, many researchers have proposed virus detection techniques based on biologically inspired techniques [9, 10, 11, 12, 13]. Most of them refer to the great ability of human immune system in protecting human body from unknown pathogen like biological viruses and propose an artificial immune system to protect the computer from computer viruses. For example, Lee, et al. [9] work on artificial immune based virus detection system that can detect unknown viruses. Their work is based on self and nonself strings defined previously in Forrest's research [13]. Other researchers [14, 15] proposed computer viral detection techniques based on artificial neural networks. Their techniques do not required signature for detecting unknown viruses. Some recently researches emphasize on detecting hard to detect metamorphic computer viruses [16, 17], introducing the term ”virus localization” [18]. The underlying principle of this research is a multiple cryptography hashing technique to locate areas within the infected file.

None of previously stated researches have suggested any approach to heal the infected code, which differ from our proposed approach.

4 Proposed Technique

From the preliminary experiment, we found that whenever the content of the message changed, the r_index would change as well. Even though the result of using indices from BWT process was quite good, these indices alone could not be used as a hash function for the integrity checking. Therefore, CRC-32 [19] are applied to supplement these indices, serving as the basis for our proposed technique.

The proposed technique consists of three processes namely, archival construction process, error detection process, and recovery process, which are described below.

4.1 Archival Construction Process

This is the first process that is responsible for rearranging, compressing, and computing necessary information for message archival purpose. As shown in Figure 2 , a message (or a file) is passed to this process where a compressed message along with the message encoder of the original message is returned. The process can be described in pseudo code as shown in Figure 3. A file and specified block size (block_size) are passed to the Transformation Module. The entire message is implicitly chopped down to N blocks. Each block is transformed by the BWT algorithm and the corresponding CRC checksum is computed. The

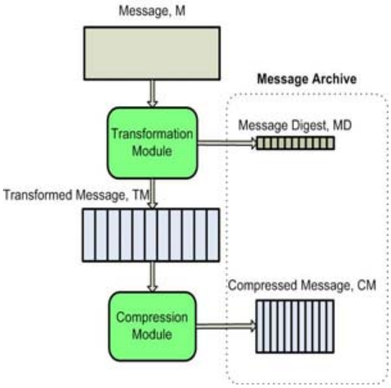


Fig. 2. Archival construction diagram

Notation

OF	Original file
MF	Message digest file
CF	Compressed file
t()	Transformation function
$(TM_i$,r_index$_i)$	Output of transformation function, the first tuple is a transformed message block the second tuple is the association index at index i
crc()	CRC checksum computation function
CRC_i	CRC checksum at index i
compress()	Compression function
M_i	Message block at index i
CM_i	Compressed Message block at index i
block_size	block size
N	number of blocks
D_i	digest block at index i
$A\|B$	| is defined as a concatenation operator

Archival Construction Process

Input : OF, block_size

```
N := sizeof(OF)/block_size
OF = $M_0M_1...M_{N-1}$
open(MF) for write
open(CF) for write
for i := 0 to N - 1 do
    ($TM_i$, r_index$_i$) := t($M_i$)
    $CRC_i$ := crc($TM_i$)
    $D_i$ := r_index$_i$ | $CRC_i$
    write(MF,$D_i$)
    $CM_i$ := compress($TM_i$)
    write(CF,$CM_i$)
end_for
```

Output : CF, MF

Fig. 3. Pseudocode for archival construction precess

output of this stage is a blockwise message digest(D, digest block), which is the combination of r_index and CRC checksum. The transformed message block is further processed by the Compression Module in Figure 2 which associates to compression function in the pseudocode. The compression function can be implemented using MTF and RLE function before it is encoded in the final state by entropy encoding techniques such as Huffman encoding or Arithmetic encoding. In each iteration, the blockwise message encoder and compressed block separately form the message digest file and compressed file, respectively.

The message block size can be arbitrary selected to discourage any guess work of the virus writers in decoding attempts. In addition, both message digest file and compressed file can be physically separated from the working file for subsequent error detection and recovery processes, making malicious decoding virtually impossible.

4.2 Error Detection Process

The main purpose of this process is to detect and locate error blocks in the message (or the file). The outputs of this process are the number of error blocks, the information to be used in decompression, and reverse transformation of the specified message block, all of which will be used in subsequent processing.

Notation

$\overline{MF}$	Message digest file of the suspected file
$\overline{D_i}$	Digest block of suspected file at index i
N_s	Number of blocks of suspected file
ErrB	Used for keep error block number and associated digest block
ErrLoc	Error locating file

Error Detection Process

Input : MF, $\overline{MF}$

```
MF = $D_0D_1...D_{N-1}$
$\overline{MF} = \overline{D}_0\overline{D}_1...\overline{D}_{N_s-1}$
open(ErrLoc)for write
if (N > $N_s$)
    for i := 0 to $N_s$ -1 do
        if ($D_i$ != $\overline{D}_1$)
            ErrB := i | $D_i$
            write(ErrLoc,ErrB)
    end_for
    for j := i to N -1 do
        ErrB := j | $D_j$
        write(ErrLoc,ErrB)
    end_for
else
    for i:= 0 to N -1 do
        if ($D_i$ != $\overline{D}_i$)
            ErrB := i | $D_i$
            write(ErrLoc,ErrB)
    end_for Output : ErrLoc
```

Fig. 4. Pseudocode for Error Detection Process

The procedures of this process are described in Figure 4. The input of this process are the message digest file of the original file and the message digest file of the suspected file. The message digest of the suspected file can be computed by using the same procedure as in archival construction process excepts that the compressed form of the suspected message is not required. The digest block of the original and suspected files will be compared one by one. If they are not equal, the block number and digest block, which is a pair of r_index and CRC checksum, will be recorded into ErrLoc file. Three possibilities to be considered of the number of digested blocks of the original file and the suspected file are greater, less, or equal.

4.3 Recovery Process

This process will recover the original message from file archive using information from the previous process. The procedure of this process is given in Figure 5. The inputs for this process are infected file, CF, and ErrLoc. The number of error blocks is retrieved from the ErrLoc file. For each iteration, reverse transformation of the uncompressed block will replace the specified error block without having to go though the entire file. Finally, the original file or message is restored. Figure 6 shows an simple example of the process.

Notation

IF	Infected File
N_{err}	Number of error blocks
Pos	Used for keep error block number
uncompress()	Uncompressing function
reverse_t()	Reverse transformation function
replace()	Replacement function
spilt()	Spilt function which return a pair of variable (a,b)

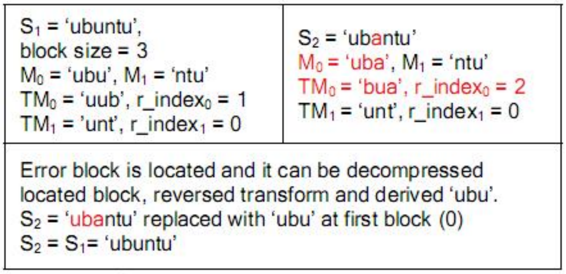
Recovery Process

Input : IF, CF, ErrLoc

```
open(ErrLoc) for read
for i := 1 to $N_{err}$ do
    ErrB := read(ErrLoc)
    (Pos,D) := spilt(ErrB)
    (r_index,CRC) := spilt(D)
    TM := uncompress(CMPos)
    M := reverse_t(TM, r_index)
    replace(IF, M, Pos)
end_for
```

Output : Disinfected file

Fig. 5. Pseudocode for Recovery Process



S_1 = 'ubuntu', block size = 3 M_0 = 'ubu', M_1 = 'ntu' TM_0 = 'uub', r_index$_0$ = 1 TM_1 = 'unt', r_index$_1$ = 0	S_2 = 'ubantu' M_0 = 'uba', M_1 = 'ntu' TM_0 = 'bua', r_index$_0$ = 2 TM_1 = 'unt', r_index$_1$ = 0

Error block is located and it can be decompressed located block, reversed transform and derived 'ubu'.
S_2 = 'ubantu' replaced with 'ubu' at first block (0)
S_2 = S_1= 'ubuntu'

Fig. 6. An example of error detecting and recovering

5 Experimental Results

The experiments were conducted in two phases, namely, the preliminary experiment and the proposed method experiment.

5.1 Preliminary Experiment

In preliminary experiment phase, the indices which derived from forward BWT were investigated to locate any discrepancies caused by content modification. We wanted to explore the pattern of change indicated by these indices as the contents were altered. The Calgary corpus [20] and four other files were selected to furnish an extensive file type coverage in the test set. Four additional files are added, consisting two Microsoft bitmap images and two unix program files, namely, bmp1.bmp, bmp2.bmp, sendmail.sendmail, and tcpdump, respectively. Numerous test sets were generated by arbitrarily selecting a pseudo random location to seed contiguous change of various sizes from 1 bit to 16 bytes in different blocking volumes. The results of the experiments are shown in Table 1. It was observed that, in most cases, as the size of seeded contiguous change increased, the indices that indicated content change also increased. Nevertheless, certain singularities remained undetected, such as similar bit patterns or coincidental computed values, etc. Such caveats were compensated by additional CRC supplement that yielded 100% correct detection.

5.2 Proposed Method Experiment

The same testing sets were tested in the error detection process that every error block can be detected. For a set of selected block sizes, the size of compressed file of file archive is shown in Table 2.

Note from Table 2 that compressibility of the original file (or message) depends primarily on file type as observed from the resulting compression ratio. Additional major benefits from the proposed approach are (1) content verification of suspicious files (or messages) can be carried out in compressed from without any decompression overhead; (2) off-line vital archives preserve the integrity of the original information, thereby easing the recovery process considerably.

Table 1. Results of using r_index as a change detection

File name	Change Detection Rate (%)				
	1 Byte	2 Bytes	4 Bytes	8 Bytes	16 Bytes
bib	45.67	49.83	71.33	89.83	97.83
bmp1.bmp	35.75	41.5	36	48.5	52.5
bmp2.bmp	58.83	59.17	62.67	66.33	67.67
book1	46.17	66.67	72.5	89.83	93.83
book2	42.17	59	73.67	89	95.83
geo	26	41.67	51.67	55.83	65.17
news	46	59	76.67	84.17	89.17
obj1	26	35	39.67	65.33	80.33
obj2	26.33	30.17	52.33	69.67	78.67
paper1	68.8	86.8	93	96.6	98.6
paper2	68.67	84.67	94.33	96.67	98.17
paper3	76.75	85	95.5	99.5	99.75
paper4	29	45.67	66.33	93.67	100
paper5	69	89.33	85.33	91.33	92.67
paper6	30.25	53.75	79	93	99.25
pic	57.67	58.33	54.17	60.17	61.17
progc	40	43.5	73	88	98.75
progl	33	58.2	73.6	86	92.4
progp	64.5	76.75	85.75	83.5	91.25
sendmail	41.83	64.5	74	79.5	89.67
tcpdump	30.17	48.83	63.67	78.33	85.5
trans	74.5	84	84.67	86.83	90.33

Table 2. Size of tested files after being compressed by BWT technique

File name	File size (Bytes)		Compression Ratio
	Original File	Compressed File	
Bib	111,261	29,567	3.76
bmp1.bmp	67,854	17,431	3.89
bmp2.bmp	1,497,206	18,944	79.03
book1	768,771	275,831	2.79
book2	610,856	186,592	3.27
Geo	102,400	62,120	1.65
News	377,109	134,174	2.81
obj1	21,504	10,857	1.98
obj2	246,814	81,948	3.01
Paper1	53,161	17,724	3
Paper2	82,199	26,956	3.05
Paper3	46,526	16,995	2.74
Paper4	13,286	5,529	2.4
Paper5	11,954	5,136	2.33
Paper6	38,105	13,159	2.9
Pic	513,216	50,829	10.1
Progc	39,611	13,312	2.98
Progl	71,646	16,688	4.29
Progp	49,379	11,404	4.33
Sendmail	3,859,419	1,375,653	2.81
Tcpdump	448,056	207,949	2.15
Trans	93,695	19,301	4.85

Table 3. List of infected files from the empirical experiment

Infected File Name	Original file size (bytes)	File size after infected (bytes)	Virus name	Virus type
setup.exe	116880	120464	Win32/Basket.A	Appending
WAVTOASF.EXE	111632	115216	Win32/Basket.A	Appending
DotNetInstaller.exe	5632	125440	Win32/BCB.A	Companion
notepad.exe	69120	8192	Win32/Belod.A	Companion
DTAC_Edge.doc	24064	28672	W97M/Deij.A	Macro(Overwriting)
smiley.doc	41472	11264	Wm/Over.A	Macro(Overwriting)
ChCfg.exe	49152	53328	Win32/Cabanas.3014.A	Appending
MRT.exe	23635392	23638545	Win32/Cabanas.3014.A	Appending
Foxit Reader.exe	5713920	6053916	Win32/HLLP.Shodi.I	Prepending
UpdatPnP.exe	128512	169984	Win32/Neshta.A	Appending
MOM.exe	49152	666,624	Win32/Muce.A	Adding Viral Code
PING.EXE	24576	24576	Win95/CIH-2563.B	Hole Cavity virus

Table 4. The summarization of virus disinfection by using proposed technique

Infected File Name	Proposed Disinfection (% of recovery)	% of file replacement	Commercial Anti-virus Software suggestion
setup.exe	100%	16%	delete or quarantine
WAVTOASF.EXE	100%	17%	delete or quarantine
DotNetInstaller.exe	100%	100%	delete or quarantine
notepad.exe	100%	100%	delete or quarantine
DTAC_Edge.doc	100%	100%	delete or quarantine
smiley.doc	100%	100%	delete or quarantine
ChCfg.exe	100%	30%	delete or quarantine
MRT.exe	100%	0.06%	delete or quarantine
Foxit Reader.exe	100%	100%	delete or quarantine
UpdatPnP.exe	100%	38%	delete or quarantine
MOM.exe	100%	100%	delete or quarantine
PING.EXE	100%	60%	delete or quarantine

5.3 An Experiment over Real Computer Viruses

A collection of computer viruses were cultured in a controlled environment. Various virus types infected on target files were analyzed, namely, overwriting virus, appending virus, prepending virus, and companion virus. Two well-known commercial anti-virus software were deployed along with the proposed technique. They are Avira Antivir Personal and ESET NOD32 Antivirus. Table 3 and

Table 4 summarize the results from real computer virus infection. Six major types of viruses were deployed, namely, appending, prepending, adding, hole cavity, overwriting, and companion viruses. Four categories of viruses that were shown to be detrimental are hole cavity, adding, prepending, and overwriting viruses. All of which required 100% replacement owing to total infection. The rest were relatively typical of virus infection with one exception, i.e., MRT.exe having 0.06% replacement. This was resulted from infection only in small number of blocks in a large file. Note that only hole cavity virus (PING.EXE) that yielded the same file size after infection. At any rate, the proposed technique successfully recovered the infected files to their original status. No commercial software could match the performance by any measures.

6 Conclusion

This research proposes a practical, yet efficient method for virus detection and virus disinfection in a message or a file. The proposed method not only is able to pinpoint the location of error, but also perform a perfect error recovery. The approach utilizes the fast BWT algorithm complemented by the CRC technique to arrive at a 100% damage repair. Moreover, the proposed method offers a number of security-tight features such as 1) off-line compressed archives of vital information 2) parameterized block size and index to preclude any illegal modifications, despite known algorithms, and 3) low computation overheads as related parameters can be made available during verification and required to be updated occasionally. We shall extend the proposed method to cover randomized error seeding and gauge the performance of our proposed method. We envision that the proposed method can be incorporated in other research and development areas, in particular, commercialization as the method is straightforward to implement on available technology.

References

1. Szor, P.: The Art of Computer Virus Research and Defense. Addison-Wesley Professional, Boston (2005)
2. Filiol, E.: Computer viruses: from theory to applications, Springer-Velag France (2005)
3. Aycock, J.: Computer Viruses and Malware. Springer, Heidelberg (2006)
4. Varney, D.: Adequacy of Checksum Algorithms for Computer Virus Detection. In: Proceedings of the 1990 ACM SIGSMALL/PC Symposium on Small Systems, March 28-30, pp. 280–282 (1990)
5. Maxwell, B., Thompson, D.R., Amerson, G., Johnson, L.: Analysis of CRC methods and potential data integrity exploits. In: Proc. Int'l Conf. Emerging Technologies, Minneapolis, MN, August 25-26 (2003)
6. Burrows, M., Wheeler, D.J.: A block sorting data compression algorithm, Tech. Report, Digital System Research Center (1994)
7. Nelson, M.: Data compression with the Burrows-Wheeler transform. Dr. Dobb's J. Softw. Tools 21(9), 46–50 (1996)

8. Ferragina, P., Giancarlo, R., Manzini, G.: The engineering of a compression boosting library: theory vs practice in BWT compression. In: Azar, Y., Erlebach, T. (eds.) ESA 2006. LNCS, vol. 4168, pp. 756–767. Springer, Heidelberg (2006)
9. Lee, H., Kim, W., Hong, M.: Artificial Immune System against Viral Attack. In: Bubak, M., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2004. LNCS, vol. 3037, pp. 499–506. Springer, Heidelberg (2004)
10. Edge, K.S., Lamont, G.B., Raines, R.A.: A Retrovirus Inspired Algorithm for Virus Detection & Optimization. In: GECCO 2006, July 8-12, pp. 103–110 (2006)
11. Kephart, J.O.: A biologically inspired immune system for computers. In: Brooks, R.A., Maes, P. (eds.) Proceedings of the Fourth International Workshop on Synthesis and Simulation of Living Systems, pp. 130–139. MIT Press, Cambridge (1994)
12. Kephart, J.O., Sorkin, G.B., Arnold, W.C., Chess, D.M., Tesauro, G.J., White, S.R.: Biologically inspired defenses against computer viruses. In: Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI 1995), Montreal, PQ, pp. 985–996. Morgan Kaufman, San Francisco (1995)
13. Forrest, S., Perelson, A.S., Allen, L., Cherukuri, R.: Self-nonself discrimination in a computer. In: Proceedings of the 1994 IEEE Symposium on Research in Security and Privacy, pp. 202–212. IEEE Computer Society Press, Los Alamitos (1994)
14. Zhang, B., Yin, J., Tang, W., Hao, J., Zhang, D.: Unknown Malicious Codes Detection Based on Rough Set Theory and Support Vector Machine. In: International Joint Conference on Neural Networks (July 2006)
15. Yoo, I.S., Ultes-Nitsche, U.: Non-signature based virus detection. Journal in Computer Virology 2(3), 163–186 (2006)
16. Webster, M., Malcolm, G.: Detection of metamorphic computer viruses using algebraic specification. Journal in Computer Virology 2(3), 149–161 (2006)
17. Wong, W., Stamp, M.: Hunting for metamorphic engines. Journal in Computer Virology 2(3), 211–229 (2006)
18. Crescenzo, G.D., Vakil, F.: Cryptographic hashing for virus localization. In: Proceedings of the 4th ACM workshop on Recurring malcode (November 2006)
19. Koopman, P.: 32-bit cyclic redundancy codes for Internet applications. In: Intl. Conf. Dependable Systems and Networks (DSN), Washington, DC, pp. 459–468 (2002)
20. The Calgary corpus may be downloaded from, ftp://ftp.cpsc.ucalgary.ca/pub/projects/text.compression.corpus

Procedurally Placement of Tropical Beach Vegetation in Level Design Tools

Pisal Setthawong, Pornchai Mongkolnam, and Vajirasak Vanijja

School of Information Technology
King Mongkut's University of Technology Thonburi
Bangkok, Thailand
{51500701,pornchai,vachee}@sit.kmutt.ac.th

Abstract. This paper explores methods for the procedural placement of shoreline vegetation based on observations from generic tropical beach shoreline vegetation which can potentially be useful in the process of the level-design of computer games systems based on such environment. As the level design process is usually the most time-consuming process of the development of computer games, this approach can potentially save the level-designer's time by automating the process of the placement of the vegetation which could later be refined by the level-designer or used directly.

Keywords: Procedural Placement, Level Design Tools, Tropical Beach Shore-Line Vegetation, Computer Game.

1 Introduction

In the computer game development process, one of the most important but time-consuming task is the process of level-design. Though the level design process itself includes mundane and repetitive tasks such as placement of game-objects, triggers, game elements, basic scripting, and other details; the end process of the level constitutes to the majority player's impression to the product [3]. The reason why the level design is important because it will determine what the game player will see, hear, feel, and interact with the game and leave a huge impression on the game player. As games are increasing in complexity, the player demands for better games are increasing, which in turn puts a larger requirement and demand on the level design process.

To deal with the increasing complexity and demands a number of approaches have been utilized. One of the simple and most effective approaches is to employ an increased number of dedicated level-designers to the project to accomplish the task. This approach can potentially give good results as more designers work on the level design process, but is quite expensive, and requires the lead designer to make sure the level designer produce consistent levels throughout the game.

To deal with the rising cost of level design, it is advantageous to utilize tools, algorithms, and programs that could potentially improve the throughput of level designers. In this paper, we will explore how to generate a tropical beach scene and propose modified tools and algorithms that will help the level designer work on those types of levels with increased efficiency and effectiveness.

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 160–169, 2009.

2 Existing Work

To help level designers, many approaches have been proposed over the years. Each of these approaches aims to solve subsets of the tasks that the level designer has to do in their work.

One approach that has been used regularly in certain games is to utilize algorithms to do procedural generation of levels [4]. The procedural generation of levels can generate some or all of the level – conserving time on the level-designer. Most of the procedural generation algorithms are geared towards Dungeon-Crawl types of games to limit the possibilities and the relationship between objects in the scene. The result could later be fine-tuned to correct mistakes from the level generation process or make refinements to the level. Results of the procedural generation of levels have been mixed, in which some games do well, whereas others have been criticized for its monotonous levels [10] – which the player correctly have guessed it was generated by the computer.

Procedural Synthesis [4] is an approach that is explored in the research and computer graphic domain that could potentially be beneficial in the creation and design of levels. Procedural Synthesis could be used to generate models and textures to change the look and feel of the level. This is highly interesting approach due to its benefits over traditional static generation of assets. However this approach is very expensive to use extensively in any game and VR systems due to the requirement of procedurally defining all the assets in the game being very high. However this is a very useful approach used in conjunction to with other methods.

The most popular approach in solving the dilemma is to implement and improve existing level design tools. Level design tools are programs that the level designer would utilize to produce the level. Level design tools generally are tied into a certain game engine technology and are usually limited to producing levels for that technology. Each tool comes with its sets of features that are different from each other [2][3][6][9].

With additional refinements and features, a level designer could produce more with less time. As the level design tools could be shared in the team, any improvement and feature could be shared by the team. Though improving the level design tools do not offer a significant improvement in the quality or savings in time required by the level design team, certain macros or tools could be used to save significant time on the level design process by automating the placement of decorative objects on level. Decorative objects in the level are usually not significant regarding with the gameplay. However without putting consideration to these decorative objects, the level may potentially look barren. A potential area of improvement is the utilization of procedural placement of game objects to help improve the visuals of the stage [12] by requiring little input from the level designer. Most level design tools from different game engines [1][6][10] come with tools that allow automated object placements in the scene. Most of these tools are rudimentary, and usually offer random placement over a dictated area such as a box/cube, circle/sphere. These placement tools can potentially improve the quality of the stage with procedurally placement of decorations, and speed up the time required to make such a level. However the typical approach of these tools are too generic, and do not generate objects in with much relational sense, and are ill-suited to generate large parts of the scene.

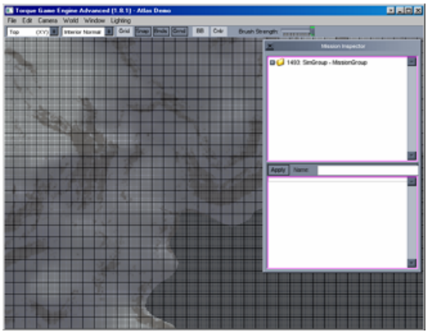

Torque Game Engine Advanced (1.8.1) - Atlas Demo
File Edit Camera World Window Lighting
Mission Inspector
1493: SimGroup - MissionGroup
Apply
Name

Fig. 1. Level Design Tool From Torque Game Engine Advanced [6]

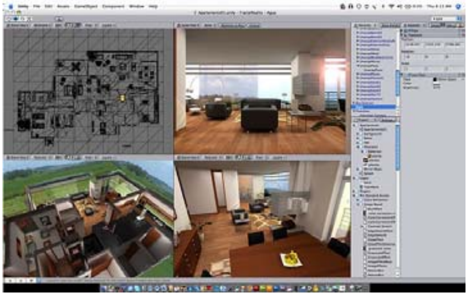


Fig. 2. Level Design Tool From Unity3D [10]

This paper would explore the procedural placement approach that is popular among game engines and extend typical placement algorithms with a more specialized version. In this paper, the generation process of generic 3D tropical beach shoreline vegetation in a level is explored.

3 Tropical Beach Shoreline Vegetation Patterns

Tropical beaches usually have quite distinct shore-line vegetation patterns. From images and illustrations of many tropical beach destinations and computer games, the common theme that reemerges is the abundance of coconut trees or related trees from the palm family. These coconut and palm trees are usually growing in abundance on patterns that cross the shoreline along the beach coastlines. In between the coconut trees usually contain other native vegetation, foliage, grass, and other smaller plants. The smaller vegetation usually grows in abundance around and around the coconut trees. The combination of the two major factors in a semi-regular pattern provides the imagery and the vegetation patterns that are associated to tropical beach shore-lines. These observations are usually based on human domain experts who later model objects and arrange them into the scene accordingly [7].

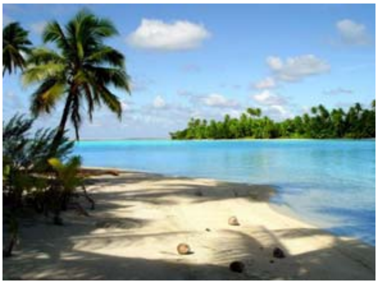


Fig. 3. Real Photograph of Coconut Trees that line the Tropical Beach Shore-Line

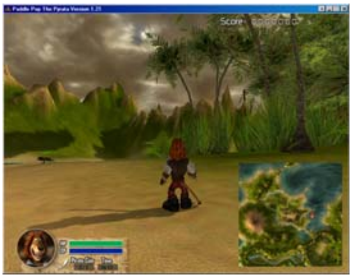


Fig. 4. PaddlePop Pyrata from CyberPlanet Interactive – an example of a game that has a level based on Tropical Beaches [9]

In the creation of such scene, the process of placement of the art assets in this tropical scene in the matter summarized earlier is usually a time consuming process due to the number of object placements that have to be done. Due to that reason, the paper explores ways to improve the placement of objects to simulate the pattern in the Tropical Beaches.

4 Proposed System

The goal of the proposed system is to product a system that procedurally places the vegetation into the scene that follows the previous discussions to generate a believable tropical shoreline beach. Based on that, the procedural placement of objects should follow the following patterns:

- The main vegetation would consist of Palm Tree family, such as Coconut Trees
- The palm/coconut trees would be placed in close proximity to the shore-line or water in a semi-clustered and semi-random function
- Though palm/coconut trees are generated in semi-clusters, it is not necessary to fill up the whole shore-line or beach with the objects
- Additional vegetations such as foliage, grass, shrubs, and other secondary vegetation could interleave or decorate areas in close proximity to the main vegetation

In our discussion we will ignore the first step of the preparation of resources for the Level Design tool and the procedural placement system. This is a trivial step and would not be discussed.

4.1 Positional Suggestion Process

The next step is to create a list of suggestion of positions where the palm/coconut trees can be positioned at. There are numerous approaches in which this could be accomplished.

One approach is to utilize contour tracing through a pre-defined paths 1. A variation of that would be to trace the beach's shorelines as a path. The contour would be defined along the boundary of the water object to the shore. With the contour, the normal and distance allowed at key control points could be used to define areas where the trees should be procedurally placed. This approach checks for shoreline boundaries directly which is good, but it is difficult to cater to unsymmetrical vegetation patterns and somewhat random patterns as the suggested positions are highly rigid.

Another approach that could be used is to allow placement to the shoreline would be the utilization of lattices. This approach can provide detailed configuration options and control to the level designer, but may be overkill as the size of the lattices and will affect the quality and the amount of the lattices to fill would take significant time.

However in our system, we would use a simpler and cleaner approach of randomly suggesting positions between a control point and a radius that forms an area of possible positional points. This allows a huge range of positional values. These values would be checked later with the **Positional Validation Test** and the **Water Proximity Test** to check for validity.

This approach allows greater control and allows easier fine-tuning of the procedural placement by moving the control point and its parameter to easily modify the resultant scene and look more natural as it is based on a semi-random fashion. The overlap of different procedural points can allow hybrid vegetation to be easily generated and influenced by the level designer. The drawback is that this approach though is that it is time-consuming as the approach lacks information of the scene, and may require excessive retries in certain scenarios.

4.2 Positional Validation Test

The next step is to create a suggestion of positions where the palm/coconut trees can be positioned at. These positions would then have to be tested with the **Positional Validation Test**. This is a simple step as the constraints to tree positions are limited by a few major factors such as the following:

- Must be positioned on the terrain and not on the water
- The terrain should not have extreme angles as palm/coconut trees do not grow on extreme angles such as on the corner of cliffs
- Must not be positions on an existing tree

If the check fails, then the position is discarded, and the process is repeated until the amount of objects are filled or the retry limit is over.

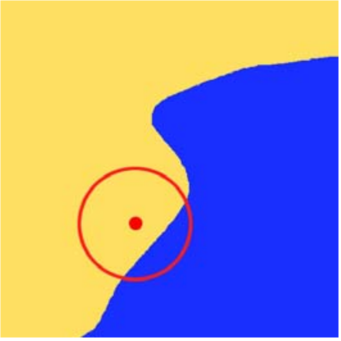


Fig. 5. Picking a Control Point and its Possible Positions Generation Area

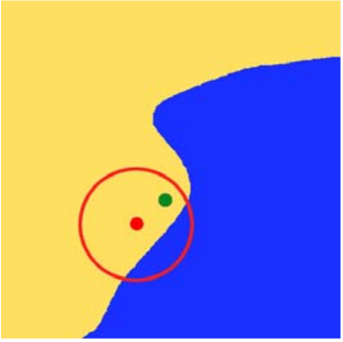


Fig. 6. A Potential Position is Selected, and Passes the Positional Validation Test

4.3 Water Proximity Test

If the suggested position is validated from the Positional Validation Test, then the next step is to perform the Water Proximity Test. Each placement would be checked to see if a water source is in the proximity of the distance allowance. This is done by doing a ray cast with a radius around the suggested position to find if there is an intersection with any water source. If the ray cast triggers a collision with a water source, then the check is validated, and the position would be added to the Valid Vegetation Placement List. If the check fails, then the position is discarded, and the process is repeated until the amount of objects are filled or the retry limit is over.

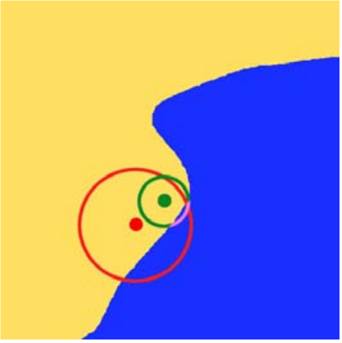


Fig. 7. The Water Proximity Test is successful due to the proximity with water, and the suggested Position is saved in The Valid Vegetation Placement List

4.4 Valid Vegetation Placement List

After the run, a list of Valid Vegetation Placement list is created based on values that follow the constraints. The primary vegetation such as the coconut/palm tree could be generated in these positions to create the base Tropical Beach vegetation. The primary vegetations are placed into the scene with varying size and alignment based on parameters that are pre-defined. These parameters are used later in randomizing the alignment of vegetation making it look more natural. In a typical case, rotation along yaw axis is usually allowed, whereas rotation along the roll and pitch axis is usually limited to a small angle.

In addition to the generation of the primary vegetation, the Valid Vegetation Placement List could be used to find positions for placing secondary vegetation such as shrubs, grass, and other additional vegetation in areas that are close proximity with the coconut/palm that are the primary vegetation by using a small displacement calculation to add with the position to generate the new position that would be used as a marker for the vegetation. The combination of the primary and sub-primary vegetation creates a visually appealing result that follows the constraints and observations that have been presented earlier.

4.5 Denseness of Vegetation

The denseness of vegetation generated is related with the object count of primary vegetation allowed in the position generation and the size of the radius of the circle where the possible positions are generated. The higher amount of objects allowed, the denser the vegetation, whereas the inverse is true. If the size of the radius that defines the possible position is decreased whereas the objects remain constant, then the vegetation would be denser, whereas the inverse is true. Factors such as adding secondary vegetation and selection of models for vegetations also changes how dense the scene would appear. The level designer can easily affect the density of the scene by modifying parameters of the system.

4.6 Overview of Proposed System

Based on discussions on the previous sub-sections, the overview of the proposed system is outlined as in the following pseudo code:

```
1. Prepare Model and Resources for the Level Design Tool
2. Select a position as the center point for where the procedural generation would generate the object and its limit radius
3. Fill in other related generation details such as ratio resizing limit, model used, amount of objects to generate, and etc.
4. Start the Placement in Proximity of Shoreline/Beach process
    4.1. Pick a random point inside the circle defined by the control point and its radius limit as long as the retry count is below the maximum allowed
    4.2. Do a Positional Validation Test to check if the selected position is not in the water. If the point is invalid, then the process is reset to step 4.1 and the retry count is increased.
    4.3. After a Positive Positional Validation test, the Water Proximity Test is done, in which the process of creating Ray Cast around the selected position is done to determine if there is water in close proximity to the object. If no water is detected in the test, the point is discarded and the process is reset to step 4.1 and the retry count is increased
    4.4. After the Water Proximity Test is passed, the position is saved in Valid Vegetation Placement List.
    4.5. The counter of successful placement is incremented
5. The Coconut/Palm trees are generated in the position that is saved in the Valid Vegetation Placement List with random size and alignment based on the parameters that are pre-defined.
6. Additional vegetation could be procedurally placed by using different placement strategies with a small displacement from the positions saved in list of Valid Vegetation Placement List by clustering the objects together
```

5 Results and Discussions

The Prototype system has been developed on a modified Torque Game Engine 1.5.2 that was modified so that it includes the features that was proposed in the previous section. As the primary author was the technical lead on the PaddlePop Pyrata project, the author reused resources from the project so that the results of the proposed procedural placement of Tropical Beach Vegetation system in the test system compared to the same level portion that was developed by a human level designer. The results from the proposed system would then be compared with a similar section of the level portion that was released in the commercial product.

A portion of the island in the level was cleared. In this section, the procedural placement of coconut tree was attempted without the modifications of the system and is illustrated in Fig. 10. The placement of the coconut trees were randomly placed over the arc. However with the placement, a number of coconut trees are placed in mountains and in positions far from the beach shorelines, which is undesirable.

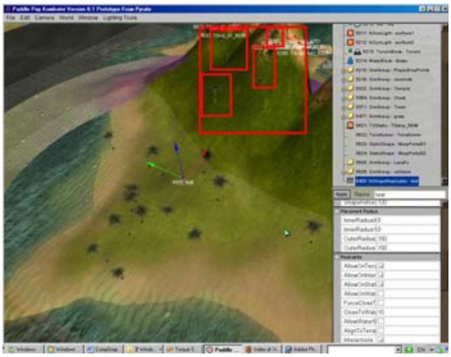


Fig. 8. Procedural Placement of Coconut Trees without Constraint Causing Anormality of Coconut Trees to Far Removed from the Shore Lines

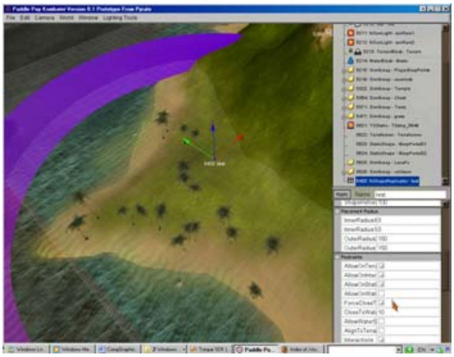


Fig. 9. Constraining Placement of Coconut Trees to Shorelines

By changing the placement strategy into the proposed strategy, where all placements where checked to its proximity to water, or if they are close to the shoreline, the coconut tree placement becomes more organized. Abnormality regarding placement of coconut trees in areas undesirable such as mountains are easily fixed by changing parameters of the influence of the distance to the beach requirements. An illustration of how the changes improve the placement is shown in Fig. 11.

To improve the system, the values from the Valid Vegetation Placement List are retrieved as markers. With these markers, a separate procedural placement strategy was used to automatically place secondary vegetation and foliage in close proximity of the primary vegetation. The placement strategy is a simple displacement vector from the marker position that checks for the object constraint to the terrain. The results of the system are displayed below along with a comparative human designed scene.

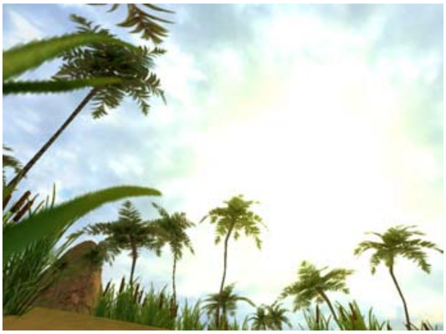


Fig. 10. Results from Proposed System

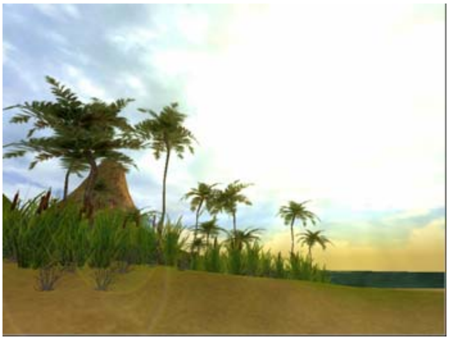


Fig. 11. Results from Proposed System

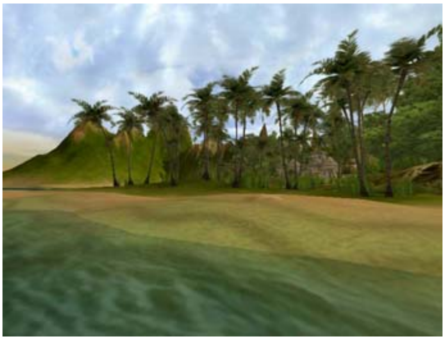


Fig. 12. Scene created by a human level designer from the Lost Jungle stage of the PaddlePop Pyrata game [9]

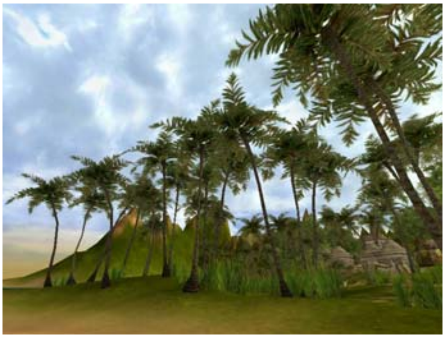


Fig. 13. Scene created by a human level designer from the Lost Jungle stage of the PaddlePop Pyrata game [9]

The proposed procedural placement produces comparable results but only requires the level designer to fill up only a few parameters cutting the placement time from hours into a few minutes.

In addition to that, a major advantage of the proposed system is in a scenario where there must be significant changes to the level. In this case, if the level was done entirely by a human level designer, then they would have to reinvest significant time to design the new level. However the proposed system could easily deal with this situation by moving the control point to the new position and changing various parameters to get the desired scene.

6 Conclusions and Future Work

The prototype system has managed to procedurally position vegetation for a virtual tropical beach that is comparable with a human level designer. The vegetation does not vary significantly between the original human designed levels on specific shorelines and could be done faster with the proposed system. However the proposed system is intended only for decoration of less-traveled or off-areas in the level, and would not be an entire replacement for a human level designer – but serve as an additional tool to utilize.

One of the drawbacks of the algorithm is that though the algorithm suggests placements of vegetation along the shoreline, due to the parameters chosen, small vegetation size, and the random nature of placement, vegetation near the shoreline may not appear dense as it should. To counter such issues, future exploration on modifying the algorithm such as implementing spline-fitting along the shoreline could be explored to find methods in generating more natural vegetation for the tropical beach.

References

1. 3AM Solutions, Dynamic VSP Object Placement, http://www.3am-solutions.com/products/dvsp3/objectPlacement.asp
2. Busby, J., Parrish, Z., VanEenwyk, J.: Mastering Unreal Technology: The Art of Level Design. Sam Publishing, Indianapolis (2005)

3. Byrne, E.: Game Level Design. Charles River Media, Boston (2005)
4. Compton, K., Michael, M.: Procedural Level Design for Platform Games. In: Laird, L., Schaeffer, J. (eds.) Proceedings of the Second Artificial Intelligence and Interactive Digital Entertainment International Conference (AIIDE), Marina del Rey, pp. 109–111 (2006)
5. Ebert, D.S., et al.: Texturing & Modeling: A Procedural Approach. Morgan Kaufmann, San Francisco (2005)
6. GarageGames,Torque Game Engine Advance, `http://www.garagegames.com/products/torque-3d`
7. Herwig, A., Paar, P.: Game Engines: Tools for Landscape Visualization and Planning. In: Buhmann, E., Nothelfer, U., Pietsch, M. (eds.) Proc.at Anhalt University of Applied Sciences, Trends in GIS and Virtualization in Environmental Planning and Design, pp. 162–171. Wichmann, Heidelberg (2002)
8. Maurina, E.F.: The Game Programmer's Guide to Torque. A K Peters, Massachusetts (2006)
9. Unilever, PaddlePop Adventure Website, `http://ww.paddlepopadventure.com`
10. Unity3D, Unity3D Engine, `http://www.unity3d.com`
11. Wallis, A.: Q&A: Flagship's Roper Talks Hellgate Pricing, Mythos, `http://www.gamasutra.com/php-bin/news_index.php?story=14524`
12. West, M.: Random Scattering: Creating Realistic Landscapes, `http://www.gamasutra.com/view/feature/1648/random_scattering_creating_.php`

Development and Performance Analysis of Grid Scheduling Algorithms

Syed Nasir Mehmood Shah, Ahmad Kamil Bin Mahmood, and Alan Oxley

Department of Computer and Information Sciences
Universiti Teknologi PETRONAS, Bandar Seri Iskandar, 31750 Tronoh, Perak, Malaysia
nasirsyed.utp@gmail.com, kamilmh@petronas.com.my,
alanoxley@petronas.com.my

Abstract. Grid scheduling is a vital component of a Grid infrastructure. Reliability, efficiency (in terms of time consumption), effectiveness in resource utilization, and robustness are the desired characteristics of Grid scheduling systems. Many algorithms have been developed for Grid scheduling. In this paper, we propose two new scheduling algorithms (the Multilevel Hybrid Scheduling Algorithm and the Multilevel Dual Queue Scheduling Algorithm) for optimum utilization of CPUs in a master/slave environment. The main idea of the proposed algorithms is to allocate jobs to cluster processors in a circular fashion and execute jobs optimally, i.e. with minimum average waiting, turnaround and response times. To facilitate the research, a software tool has been developed which produces a comprehensive simulation of a number of CPU scheduling algorithms for a clustered system. The tool's output is in the form of scheduling performance metrics.

Keywords: Distributed systems, Grid computing, Grid scheduling, load balancing, task synchronization, parallel processing.

1 Introduction

Let us begin by describing some terms. 'Scheduling' is described by the Grid Scheduling Dictionary Project as follows: "The process of ordering tasks on compute resources and ordering communication between tasks. Also, known as the allocation of computation and communication over time" [1].

Grid scheduling presents several challenges that make the implementation of practical systems a very difficult problem. Our research aims to develop a Grid scheduler that makes efficient utilization of resources and possesses a high degree of abstraction in inter-task synchronization.

The structure of the paper will now be described. Section 2 is a literature review of Grid scheduling methodologies. Section 3 describes new scheduling algorithms and section 4 shows the homogenous implementation of new algorithms. In section 5, the scheduling simulator's design and development are discussed. Section 6 shows the experimental setup and section 7 gives results and a discussion. Section 8 concludes the paper.

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 170–181, 2009.

2 Related Research

A number of factors should be considered for effective Grid scheduling, e.g. resource utilization, job allocation, load balancing, response time, inter-task dependencies, heterogeneity and scalability.

In [2] the authors proposed a middleware framework for Grids. This framework is based on an advanced reservation mechanism. It satisfies the user by providing QoS assurance for Grid applications. This is a cost effective solution for the efficient utilization of resources. The framework is also a robust one because it handles uncertain runtimes of applications intelligently. However, there are shortcomings with this framework. Firstly, it has not been integrated into the Grid toolkit and so no real-time performance tests have been undertaken. Secondly, it works on pre scheduling mechanism. Integration with Grid toolkit will facilitate for the development and deployment of QoS enabled Grid systems.

In Grid scheduling, some means of estimating a task's execution time must be used. Furthermore, information about each node's capability and availability must be gathered. The matching of tasks to nodes and the monitoring of the tasks needs to take place. The software to perform these management functions could be located on a central computer, i.e. centralized, or could be located on several computers, i.e. decentralized [3, 4].

Each node of a Grid has its own local scheduling policy. When some nodes apply their priority policies in favor of local jobs, then global jobs making use of these nodes will suffer from much longer response times. As a result, the overall performance of the Grid will be degraded. In [5] the authors propose an adaptive site selection algorithm for a Grid scheduler based on the priority policies of local schedulers. The experimental results show that the proposed algorithm can lower the difference in average waiting times among the sites with different priority based scheduling policies. The proposed algorithm maintains a Remote Queue and a Local Queue at each node of the Grid. The drawback with this algorithm is that too much processing time is involved in accessing the large number of queues at the distributed nodes.

In [6] the authors propose a dynamic task scheduling technique based on the "divide and conquer" principle. The proposed technique is dynamic, mixed, non-preemptive, adaptive and fully distributed. The contribution of this technique is its approach to transfer and placement decisions. There is a deficiency, however, in that no work has been done regarding task dependencies.

In [7] the authors propose a Grid scheduling system for the processing of complex tasks in a Grid. This scheduling system makes use of user data entry. The user inputs descriptions of the tasks' computational and data-related requirements. This approach works efficiently for the execution of dependent tasks. Access and management of resources should be done via a dedicated API. The authors mentioned that currently no such API is available and therefore its development is an open challenge for researchers.

[8] proposes a compensation based scheduling approach to Grid scheduling. This approach provides predictable execution times by monitoring Grid application performance. This approach compares the monitored application performance with the desired application performance. This approach also performs corrections by dynamically allocating additional resources. This approach has been implemented and

evaluated using the ALiCE Grid system. Its scalability has been studied using a simulation. Experimental results show that compensation based scheduling is effective in reducing execution time estimation misses and the total execution times of Grid applications. The authors also highlighted future work, which includes multi-resource compensation, resource partitioning and allocation, the improvement in the execution time estimator, and the use of heuristics and dynamic methods to determining the value of a sensitivity factor in the application execution rate formula. The weakness of this scheduling approach is that task dependencies have not been considered.

In [9] the authors investigated the performance effects of delays in propagating information about computing node failure. Generally, Grid scheduling is performed by global or external schedulers, i.e. remotely from computing nodes. However, if a node fails then it has no mechanism to communicate this information to the Grid scheduler. A Grid scheduling system should be robust enough to deal with uncertainty. The authors pointed out that none of the current studies adequately deal with the consequence of failure at the resource level. Only a few scheduling systems have been developed with a fault tolerant perspective. In [9] the authors studied several schedulers having a range of delays in transmitting node failure information. However, no new idea was proposed.

In [10] the author proposed an adaptive scheduling system by using a Max-min algorithm. The experimental results show that the proposed model can schedule tasks efficiently. The proposed system is particularly good at detecting and using idle processors. This system dynamically selects the proper scheduling strategy according to the accuracy of the predictor. This system also considers the dynamic characteristics of Grid applications and also makes the scheduling adaptive to the Grid environment.

An 'Ant algorithm' has been proposed for efficient resource management and task scheduling in Grids [11]. An Ant algorithm is a type of heuristic algorithm. Such algorithms have existed for several decades. (A related algorithm, the Ant Colony Optimisation algorithm, is currently receiving much attention.) In our context, the algorithm includes a resource state prediction mechanism for proper dynamic task scheduling. It also has the inherent parallelism and scalability features which make it very suitable for dynamic task scheduling. A simulator was designed and developed to validate the scalability of the Ant algorithm. Simulation results showed that the algorithm performs well as regards response time, resource average utilization and task parallel proportion. A shortcoming with this algorithm is that it has not been tested on a real time Grid environment.

In [12] the authors proposed a resource management and scheduling model based on a hybrid approach containing both a genetic algorithm and an ant algorithm. They simulated the algorithm by using the SimGrid toolkit. They compared the results with several typical Grid scheduling strategies. The concluded results showed that the model affords good expandability and load balancing.

3 New Scheduling Algorithms

In [13] we proposed two scheduling algorithms – the Hybrid scheduling algorithm and the Dual Queue scheduling algorithm. For completeness, each of these will now be described.

3.1 Hybrid Scheduling Algorithm (H)

For the H algorithm the ready queue is maintained in order of CPU burst length, with the shortest burst length at the head of the queue. Two numbers are maintained. The first number, t_{large} , represents the burst length of the largest PCB in the queue while the second one, t_{exec} , represents a running total of the execution time of all processes (since a reset was made). A PCB of a process submitted to the system is linked to the queue in accordance with its CPU burst length.

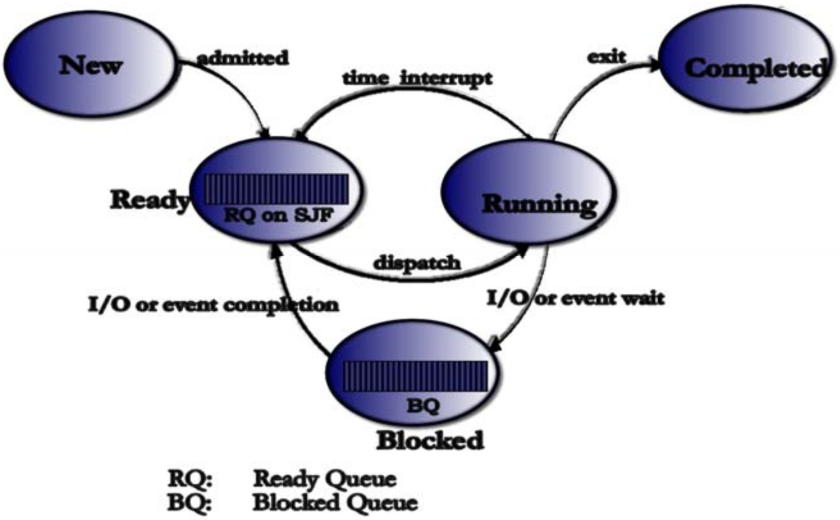


Fig. 1. Process State Diagram of H

H dispatches processes from the head of the ready queue for execution by the CPU. Processes being executed are preempted on expiry of a time quantum, which is a system-defined variable. Following preemption, t_{exec} is updated as follows:

$t_{exec} = t_{exec} + quantum$

The numbers are then compared.

If $t_{exec} < t_{l\arg e}$ then the preempted process's PCB is linked to the tail of the ready queue. The next process is then dispatched from the head of the ready queue.

If $t_{exec} \geq t_{l\arg e}$ then the PCB with the largest CPU burst length is given a turn for execution. Upon preemption, the ready queue is sorted on the basis of SJF.

The value of t_{large} is reset to the burst length of the largest PCB, which is lying at the tail of the queue, and t_{exec} is reset to 0. The next process is then dispatched from the head of the ready queue.

When a process has completed its task it terminates and is deleted from the system. t_{exec} is updated as follows:

$t_{exec} = t_{exec} + \textit{time to complete}$

The numbers are then compared and the actions taken are the same as those for a preempted process.

3.2 Dual Queue Scheduling Algorithm (DQ)

For the DQ algorithm the ready queue comprises two queues – the waiting queue and the execution queue. The waiting queue is maintained as a FIFO queue. A PCB of a process submitted to the system is linked to the tail of the waiting queue. Whenever the execution queue is empty, all PCBs in the waiting queue are moved to the execution queue, leaving the waiting queue empty. The execution queue is maintained in order of CPU burst length, with the shortest burst length at the head of the queue. Two numbers are maintained. The first number, t_{large} , represents the burst length of the largest PCB in the ready queue (waiting queue and execution queue combined) while the second one, t_{exec} , represents a running total of the execution time of all processes (since a reset was made). The algorithm dispatches processes from the head of the execution queue for execution by the CPU. Processes being executed are preempted on expiry of a time quantum, which is a system-defined variable. Following preemption, t_{exec} is updated as follows:

$t_{exec} = t_{exec} + \ quantum$

The numbers are then compared.

If $t_{exec} < t_{l\arg e}$ then the preempted process's PCB is linked to the tail of the execution queue. The next process is then dispatched from the head of the execution queue.

If $t_{exec} \geq t_{l\arg e}$ then the PCB with the largest CPU burst length is given a turn for execution. Upon preemption, all PCBs in the waiting queue are moved to the

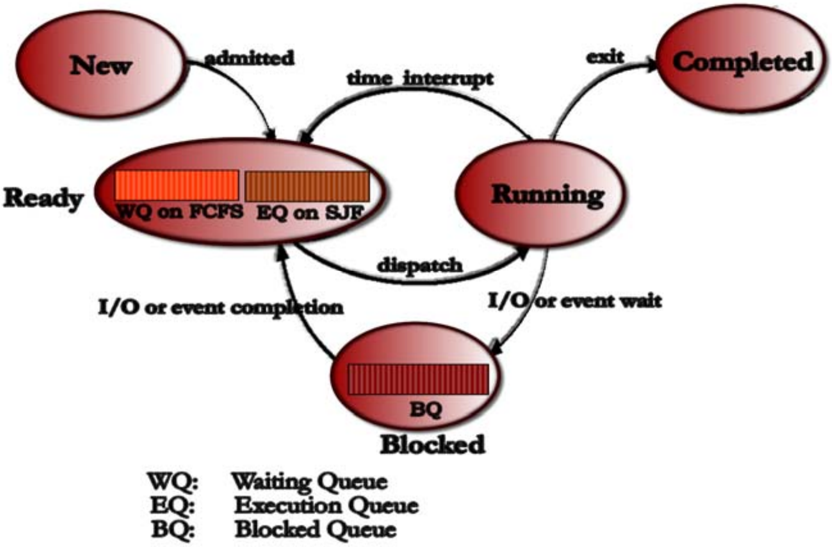


Fig. 2. Process State Diagram of DQ

execution queue, leaving the waiting queue empty. The execution queue is then sorted on the basis of SJF. The value of t_{large} is reset to the burst length of the largest PCB and t_{exec} is reset to 0. The next process is then dispatched from the head of the execution queue.

When a process has completed its task it terminates and is deleted from the system. t_{exec} is updated as follows:

$t_{exec} = t_{exec} +$ *time to complete*

The numbers are then compared and the actions taken are the same as those for a preempted process.

4 Homogeneous Implementation of New Algorithms

In [13] we proposed two scheduling algorithms – the Hybrid scheduling algorithm and the Dual Queue scheduling algorithm. For completeness, each of these will now be described.

Let us consider the implementation of one of these new algorithms throughout the nodes of a cluster. One type of cluster architecture is the master/slave architecture as shown in Fig.3. The cluster takes process sets as input and distributes processes on cluster processors using round-robin scheduling (i.e. 1, 2, 3…. n, 1) for parallel computation by the slaves.

The same algorithm, either H or DQ, is used on each slave processor. Once a computation is complete, the results are sent to the master processor.

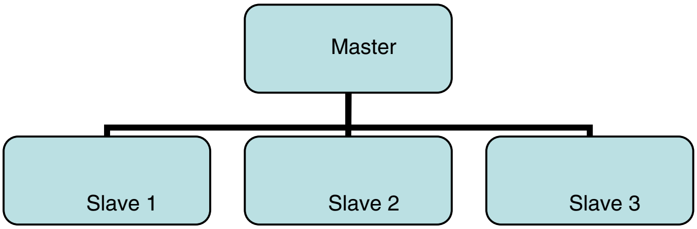


Fig. 3. Block diagram of master/slave architecture

5 Scheduling Simulator Design and Development

In order to evaluate the effectiveness of our approach, a scheduling simulator was developed. This involves the use of an actual cluster. The simulation software comprises two parts. One part runs on the Master node (SimM). A copy of the other part runs on each slave (SimS). The user of SimM inputs metadata about hypothetical jobs. The metadata for each job includes its ID, its size, its priority and the number of slaves that the job is to be divided between. The user of SimM also inputs a scheduling policy that is to be used by the slaves. SimM distributes the metadata to the slaves. SimM receives notification from each slave for each job (or part of a job) that has finished. Let us now turn our attention to the slaves. Each slave runs SimS. SimS is notified by SimM of the scheduling policy to be used by all slaves. It should be noted that there are no jobs, as such, only hypothetical jobs. Only job metadata is passed to SimS. SimS processes the metadata for the list of processes that have been assigned to it. Upon completion of a process, SimM is informed. No 'useful' work is done by a slave other than that associated with scheduling. SimS keeps a detailed record of the processes being run on the slave - process ID; CPU burst length; arrival time; time quantum; priority.

All slaves use the same CPU scheduling algorithm, which is input by the user of SimM. The user can select one of a range of algorithms including the newly developed ones, MH and MDQ, as well as established ones, FCFS, SJF, SRTF, RR, and the Priority scheduling. The purpose of the simulator is to produce a comparative performance analysis of CPU scheduling algorithms. The simulator is written in Java and makes use of the 'MPJ express' API.

6 Experimental Setup

All scheduling algorithms have been tried with several synthetic workloads. The results have been evaluated. The experiments made use of a HPC facility in the Department of Computer and Information Sciences at Universiti Teknologi PETRONAS. We ran our experiment using a cluster of 8 processors. The hpc.local was used as the default execution site for job submission. A detailed experimental setup is shown in Table 1.

Table 1. Experimental Setup

Name	Type	Location	Configuration
nasir	Shell terminal	FYP Lab Workstation	Intel Core 2 Duo CPU 2.0GHZ 2 GB Memory
hpc.local	Execution site	HPC facility	CPUs: 4x2.3 GHZ Memory: 7.75GB 483.427GB
compute-0-0.local	Execution site	HPC facility	CPUs: 4x2.3 GHZ Memory: 7.80GB 483.427GB

(a) Small size workload 100-200 Processes

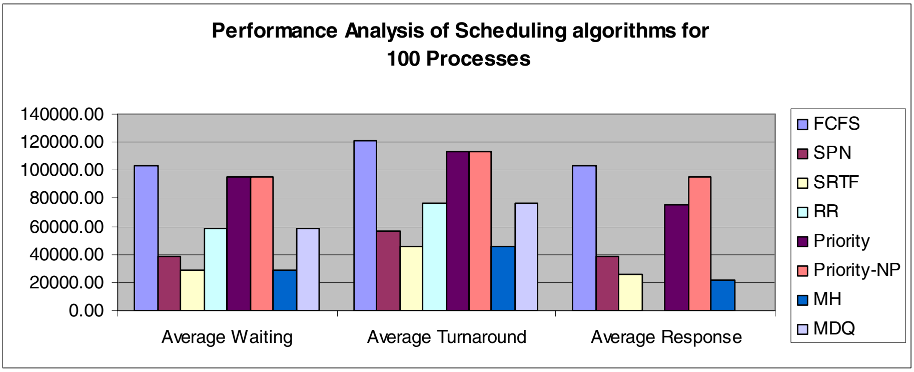


Fig. 4. Comparative Performance Analysis for 100 Processes

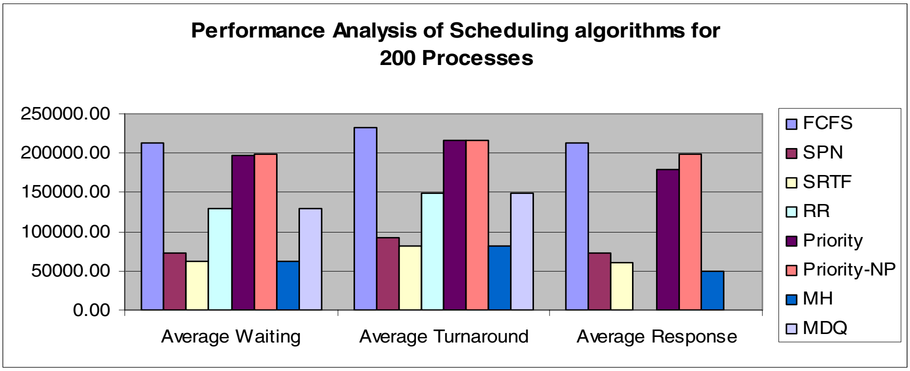


Fig. 5. Comparative Performance Analysis for 200 Processes

(b) Medium size workload 500-1000 Processes

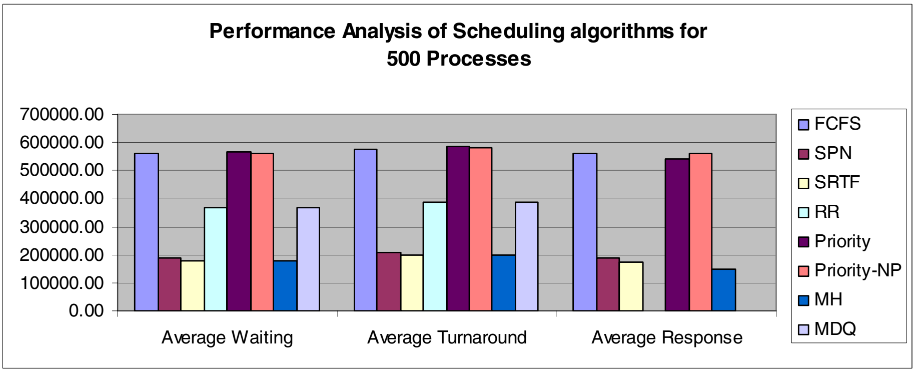


Fig. 6. Comparative Performance Analysis for 500 Processes

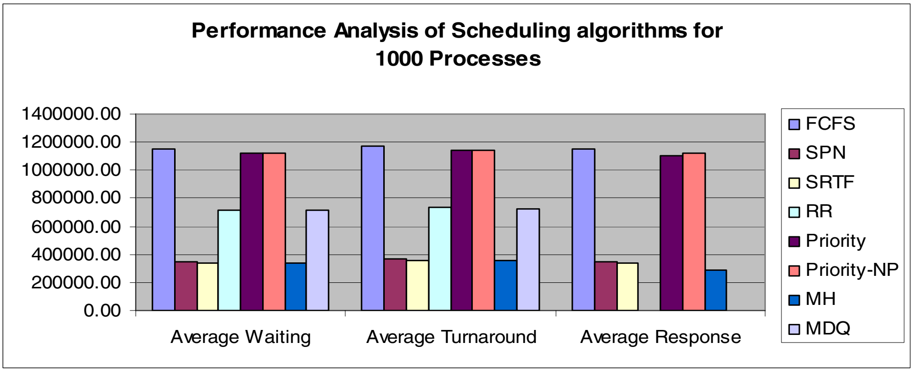


Fig. 7. Comparative Performance Analysis for 1000 Processes

(c) Large size workload 1500-2000

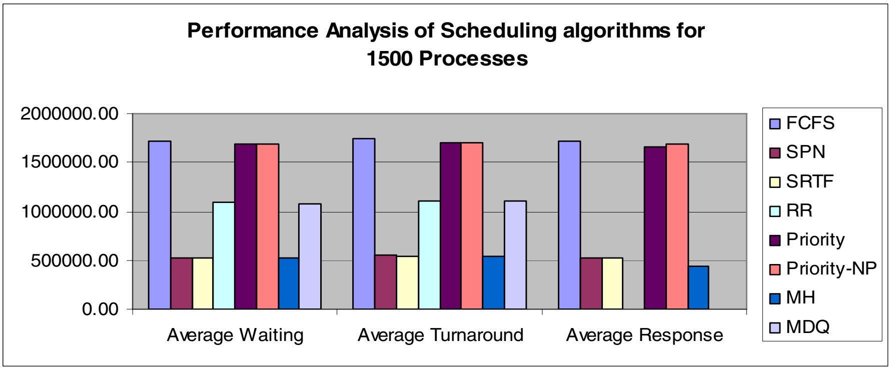


Fig. 8. Comparative Performance Analysis for 1500 Processes

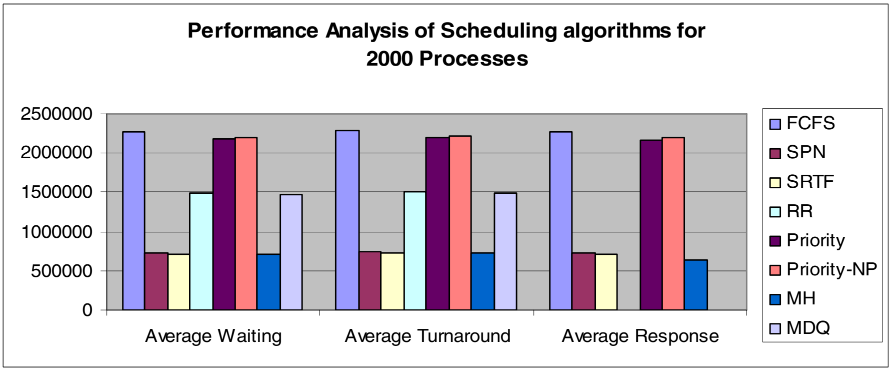
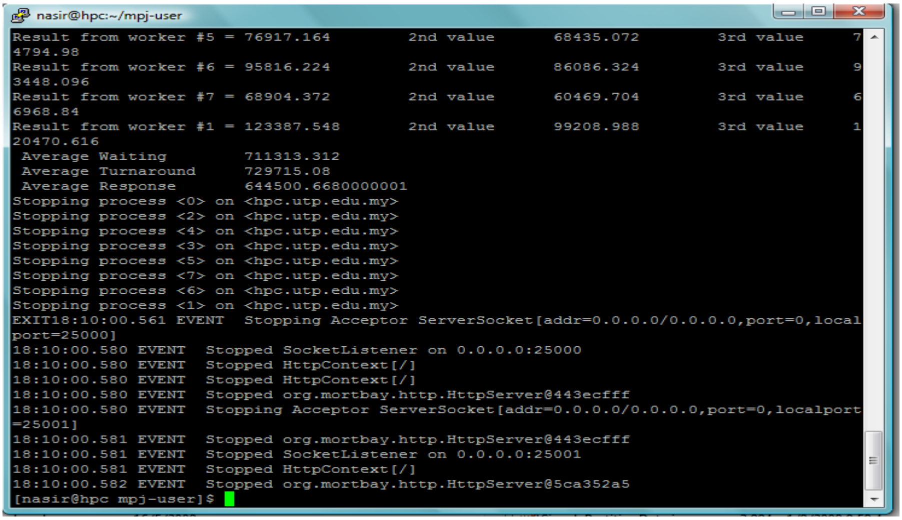


Fig. 9. Comparative Performance Analysis for 2000 Processes



```
Result from worker #5 = 76917.164       2nd value       68435.072        3rd value       7
4794.98
Result from worker #6 = 95816.224       2nd value       86086.324        3rd value       9
3448.096
Result from worker #7 = 68904.372       2nd value       60469.704        3rd value       6
6968.84
Result from worker #1 = 123387.548      2nd value       99208.988        3rd value       1
20470.616
 Average Waiting        711313.312
 Average Turnaround     729715.08
 Average Response       644500.6680000001
Stopping process <0> on <hpc.utp.edu.my>
Stopping process <2> on <hpc.utp.edu.my>
Stopping process <4> on <hpc.utp.edu.my>
Stopping process <3> on <hpc.utp.edu.my>
Stopping process <5> on <hpc.utp.edu.my>
Stopping process <7> on <hpc.utp.edu.my>
Stopping process <6> on <hpc.utp.edu.my>
Stopping process <1> on <hpc.utp.edu.my>
EXIT18:10:00.561 EVENT  Stopping Acceptor ServerSocket[addr=0.0.0.0/0.0.0.0,port=0,local
port=25000]
18:10:00.580 EVENT  Stopped SocketListener on 0.0.0.0:25000
18:10:00.580 EVENT  Stopped HttpContext[/]
18:10:00.580 EVENT  Stopped HttpContext[/]
18:10:00.580 EVENT  Stopped org.mortbay.http.HttpServer@443ecfff
18:10:00.580 EVENT  Stopping Acceptor ServerSocket[addr=0.0.0.0/0.0.0.0,port=0,localport
=25001]
18:10:00.581 EVENT  Stopped org.mortbay.http.HttpServer@443ecfff
18:10:00.581 EVENT  Stopped SocketListener on 0.0.0.0:25001
18:10:00.581 EVENT  Stopped HttpContext[/]
18:10:00.582 EVENT  Stopped org.mortbay.http.HttpServer@5ca352a5
[nasir@hpc mpj-user]$ 
```

Fig. 10. Cluster Snapshot - Performance Metrics of MH for 2000 Processes

Table 2. Performance Analysis of Scheduling Algorithms for 2000 Processes

	Performance Factors		
Scheduling Algorithm	**Average Waiting Times**	**Average Turn- around Times**	**Average Response Times**
FCFS	2263769.72	2282171.48	2263769.72
SJF	727082.92	745484.69	727082.92
SRTF	711183.92	729585.69	704514.30
Priority	2186655.12	2205056.89	2167650.37
Priority(NP)	2195569.54	2213971.31	2195569.54
RR	1481041.75	1499443.52	1333.29
MH*	711313.31	729715.08	644500.67
MDQ*	1469092.90	1487494.66	4630.54

7 Results and Discussion

To check the performance of the proposed algorithms, i.e. MH and MDQ, we made use of a number of CPUs and synthetic workloads. Each workload had different characteristics.

Performance metrics for the CPU scheduling algorithms are based on three factors - Average Waiting Time, Average Turnaround Time, and Average Response Time. Table 2 shows the average time, for 2000 processes, for each performance factor and for each algorithm.

Figs. 4 to 9 show that the relative performance of the CPU scheduling algorithms is independent of the workload size. MH is based on RR and SJF. Fig. 9 shows that average waiting and turnaround times for MH are comparable to SPN and SRTF but shorter than those for all other CPU scheduling policies. Furthermore, Fig. 9 shows that the average response time for MH is longer than the values for RR and MDQ, comparable to SPN and SRTF, but shorter than those for the other CPU scheduling policies. There is no starvation for any jobs when using the MH. There is a little overhead due to the sorting involved with MH.

MDQ is an extended form of MH. Fig. 9 shows that the average waiting time for MDQ is longer than that for MH. Overall, three CPU scheduling policies have shorter least average waiting times than MDQ, three longer, and one about the same. Similarly, three CPU scheduling policies have shorter least average turnaround times than MDQ, three longer, and one about the same. Furthermore, Fig. 9 shows that the average response time for MDQ is to that for RR. They produce shorter average response times than all other CPU scheduling policies. MDQ gives good results, especially response time, with little overhead for all nature of process sets. Another advantage of MDQ, as stated earlier, is that there is no starvation.

8 Conclusion and Future Work

In this paper we present two new scheduling algorithms. We have evaluated these algorithms on a simulator running on a Cluster. We compared the efficiency of our algorithms with six other well known CPU scheduling algorithms. MH works well from a system perspective. We can say that MH is a scheduling policy from the system point of view; it satisfies the system requirements (i.e. short Average Waiting Time and short Turnaround Time). MDQ works well from the user perspective due to its short Average Response Time). Moreover, MH and MDQ are scalable, i.e. the relationship between each performance measure (e.g. average waiting time) and the workload size is very nearly linearly related.

In future, we will enhance our scheduling algorithms for use on a GRID computing environment. We will develop new methods of inter-task synchronization and robustness to integrate with our Grid scheduling algorithms.

References

1. Grid Scheduling Dictionary Project, `http://www.mcs.anl.gov/~schopf/ggf-sched/GGF5/sched-Dict.1.pdf`
2. Farooq, U., Majumdar, S., Parsons, E.W.: Achieving efficiency, quality of service and robustness in multi-organizational Grid. The Journal of Systems and Software (2008)
3. Dhodhi, M.K., et al.: An integrated technique for task matching and scheduling onto distributed heterogeneous computing systems. J. of Parallel and Distributed Computing 62, 1338–1361 (2002)
4. Lee, S.Y., Cho, C.H.: Load balancing for minimizing execution time of a target job on a network of heterogeneous workstations. In: Feitelson, D.G., Rudolph, L. (eds.) IPDPS-WS 2000 and JSSPP 2000. LNCS, vol. 1911, pp. 174–186. Springer, Heidelberg (2000)
5. Wiriyaprasit, S., Muangsin, V.: The Impact of Local Priority Policies on Grid Scheduling Performance and an Adaptive Policy-based Grid Scheduling Algorithm. In: Proceedings of the Seventh International Conference on High Performance Computing and Grid in Asia Pacific Region (2004)
6. Savvas, I.K., Kechadi, M.T.: Dynamic Task Scheduling in Computing Cluster Environments. In: Proceedings of the ISPDC/HeteroPar 2004 (2004)
7. Plantikow, S., Peter, K., Högqvist, M., Grimme, C., Papaspyrou, A.: Generalizing the data management of three community Grids. Future Generation Computer Systems (2008)
8. Teo, Y.M., Wang, X., Gozali, J.P.: A Compensation-based Scheduling Scheme for Grid Computing. In: Proceedings of the Seventh International Conference on High Performance Computing and Grid in Asia Pacific Region (2004)
9. Thomas, N., Bradley, J., Knottenbelt, W.: Performance of A Semi Blind Service Scheduler. In: Proceedings of the UK e-Science All Hands Meetings, Nottingham (2004)

10. Lee, L.T., Liang, C.H., Chang, H.Y.: An Adaptive Task Scheduling System for Grid Computing. In: The Sixth IEEE International Conference on Computer and Information Technology (2006)
11. Xu, Z., Hou, X., Sun, J.: Ant Algorithm-based Task Scheduling in Grid Computing. In: IEEE CCECE 2003, Electrical and Computer Engineering,, vol. 2, pp. 1107–1110 (2003)
12. Tian, H.: A New Resource Management and Scheduling Model in Grid Computing Based on a Hybrid Genetic Algorithm (2008)
13. Shah, S.N.M., Mahmood, A.K.B., Oxley, A.: Hybrid Scheduling and Dual Queue Scheduling. In: 2nd International Conference on Computer Science and Information Technology, Beijing, China (2009)

Association Rule Mining for Intellectual Capital of Enterprises in Central Region of Thailand

Anongnart Srivihok

Department of Computer Science, Faculty of Science, Kasetsart University,
Bangkok 10900, Thailand
fsciang@ku.ac.th

Abstract. Intellectual Capital (IC) is considered as an intangible asset of an organization and further it is used as the measure of organizational assets in some organizations in Europe and Australia. Nevertheless, the study of IC was not extensively investigated in Thailand. The objective of this paper was to apply a data mining technique, association rule to mine data collected from target organizations. In this study, public and private organizations located in the central part of Thailand were surveyed on their Intellectual Capital characteristics. These target organizations were divides into three classes: low, average, high according to their IC scores. Then, association rule algorithm, Apriori was applied to find the relationships of 24 attributes in each IC Class. Results of association rule mining were reported. The implication of this model was also suggested.

Keywords: Intellectual Capital, association rules, Apriori algorithm, data mining.

1 Introduction

Thailand with the vision towards knowledge based economy has transformed an importance of physical assets to highlights of intangible high value added products and services such as software development, designed products, scientific and financial consulting services. These intangible assets are defined as intellectual capital.

Intellectual Capital can be measured by deducting an organization's book value (i.e. the value of physical assets reported by standard accounting practices) from market value [1]. That is the market values are the sum of financial capital (tangible capital) and intellectual one (intangible capital). IC measurements are performed in some enterprises in order to predict their competitiveness in the future and to foresee the transparency of value construction and management [4].

2 Background of the Study

Intellectual Capital may be used interchangeably with intangibles, knowledge or knowledge resources [9]. There are various definitions of Intellectual Capital (IC) since IC characteristics are invisible and dynamic. Different researchers have defined IC in different definitions. Wiig [10] defines IC as "assets created through

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 182–190, 2009.

intellectual activities ranging from acquiring new knowledge (learning) and inventions to creating valuable relationships" and IC is defined as "the difference between a company's market value and its book value" [8]. The most famous IC definition has been proposed by Edvinsson [5] which states that "The Intellectual Capital of a firm is its possession of the knowledge, applied experience, organizational technology, customer relationships and professional skill that provides it with a competitive edge in the market". Edvinsson [5] proposed the Skandia Intellectual Model which is extensively referred to IC measurement and research. In this model, IC is comprised of human capital and structural capital. Human Capital includes knowledge, know-how, skills and personnel expertise of an enterprise. Structural Capital is a composite element that includes organizational capital and customer capital. Organizational capital consists of innovation capital (intellectual property and intangible assets) and process capital (databases and information systems). Customer capital is the external capital which includes the organizational relationships with external actors including customers, suppliers, partners and/or other stakeholders [6,7]. Figure 1 depicted Intellectual Capital model proposed by Edvinsson [5].

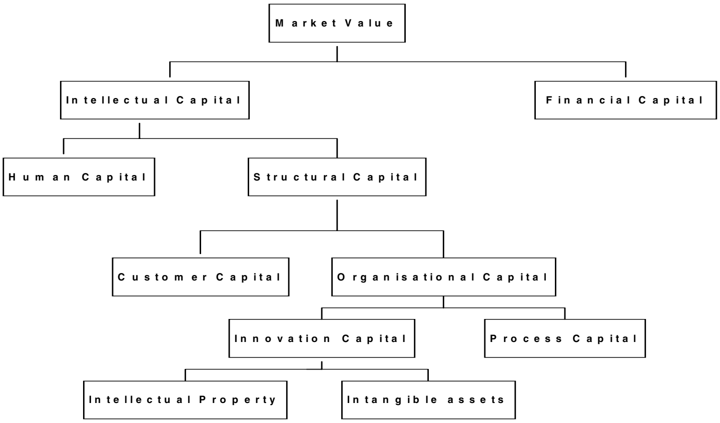


Fig. 1. Intellectual Capital model by Edvinsson

In Thailand, the multiple criteria decision-making (MCDM) approach was applied to measure IC of Thai SME [2]. Intellectual capital includes human capital, structural capital and relational capital. The IC of the ten case studies of SMEs was investigated and measured at the strategic level. Further, the ten companies might use the end results to improve their levels of IC via the three main components of human, structural, and relational capital in order to increase their competitiveness and retain their sustainable development.

3 Empirical Study

The research was designed to be a cross sectional, explanatory statistical, and mail survey. The survey study investigated in Thai public and private organizations.

Respondents of the survey were staff in the organizations, and the survey focused on measuring attributes of Intellectual Capital as shown in Table 1 and Figure 2. More details on the questionnaire can be accessed from a web site: www.smexpert.kasetsart.org/ric. It was proposed that Intellectual Capital was related to human capital, structural capital and relational capital which would affect the organization's performances. For this purpose, Intellectual Capital would be measurable by 24 attributes of human, structural and relational capitals [3].

Table 1. Twenty four attributes of Intellectual Capital used in association rule mining

IC Components	Attributes
Human Capital: Staff Competency	% staff with knowledge in working % staff using internet % experienced staff
Human Capital: Competency Improvement	Organizational support for a learning organization % staff get appropriate training % staff apply acquired knowledge in working
Human Capital: Staff Structure	% staff with long working years ability to replace a staff staff satisfaction to manager
Human Capital: Staff Stability	Good work environment % staff turnover
Structural Capital: Production technology and IT diffusion	% computers / staff % IT investment Organization database
Structural Capital: Business Philosophy	Investment in planning and implementation of the plan Organization defines the missions clearly Customer oriented organization
Structural Capital: Organizational Structure	Organization structure (2, 3 or 4 levels)
Structural Capital: Intellectual Property	Number of Intellectual Properties % research and development expenses / revenue
Relational Capital: Customer base	% customer satisfaction % customer loss
Relational Capital: Market Proximity	number of communication channels in organizations (internet, phone, fax, mail, direct) % number of communication channels in organizations

Fig. 2. shows the Intellectual Capital model using from this study which obtained from the study of Srivihok [3].

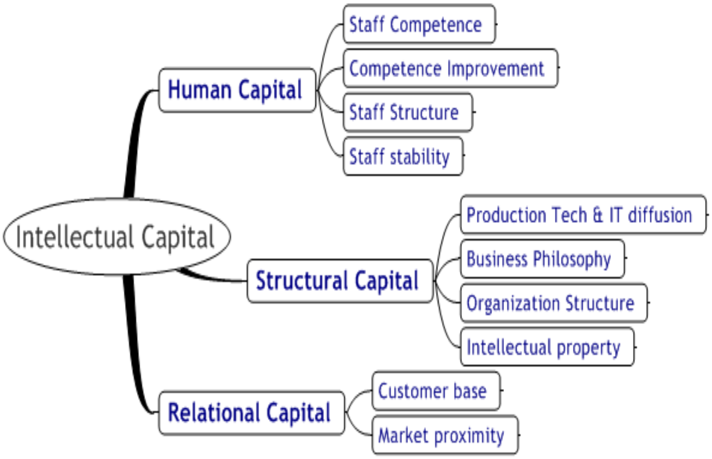


Fig. 2. Intellectual Capital model using in this study (Srivihok [3])

3.1 Data Collection

The study of Intellectual Capital measurement was focused in both public and private organisations in Thailand. There were two parts of questionnaires: intellectual capital section and organisational background. The responses in the first section were measured on a 10-point semantic differential scale with 1= strongly disagree, and 10 = strongly agree. The first section contained questions for attribute evaluation. The background information section was designed to obtain information on organisation characteristics including type of industry, size, and years of services. The first part of the questionnaire was developed from the IC model proposed by Srivihok [5] as shown in Table 1. Data collection was conducted by mail survey to 800 public and private organisations. Names and addresses of private organisations were obtained from the list of Import-Export companies provided on the website of the Department of Extension, Ministry of Commerce, Thailand. The list of public organisations was obtained from government agencies. There were about 216 respondents from the survey which was about 27%. This rate was considered adequate for a mail survey.

For the organisational characteristics, majority of them were SME, about 42.9% were small enterprises with less than 50 staff, 23.8% were medium enterprises with 51-200 staff, and 20.5% and 12.8 % were large enterprises with 201-1000 staff and more than 1000 staff, respectively. The industry type and number of participants were as follows. The majority of type of industry was manufacturing (32.60%), the next was education 14.8% and the smallest was tourism which was about 1.00%. For the organizational type, about 24.8% were government, 3.6% were state enterprises and 71.6% were private enterprises which were the majority of the population of this study.

3.2 Data Analysis

WEKA software Version 3.5.6 (Waikato Environment for Knowledge Analysis) [13] was used to analyze the data from the survey. Cross-validation of data set was conducted by using the Hold-Out Method. The data set was divided into two sub-sets: training set and testing set. The training set consisted of 90% of the total data while the remaining 10% was employed in the testing set. The data mining algorithm used was association rule, Apriori algorithm. There were 216 instances and 24 attributes were analyzed by this algorithm. The IC model was developed on the basis of this algorithm and was used to predict the relationships of attributes of organisations.

3.3 Data Mining Techniques

Association analysis is used for finding relationships of features in large database. It is familiar for predicting or analyzing the customer purchase called "Market Basket Analysis". This task is performed by using Association Rule mining algorithm to compose the problem into subtasks. For example of a grocery store, the customers always buy (1) bread and milk (2) bread, diapers, beer and eggs (3) milk, diapers, beer and Cola (4) bread, milk, diapers, and beer and (5) bread, milk, diapers, and Cola. By applying association rules, it is suggested that a strong relationship between the sales of diapers and beer, because many customers who buy diapers also buy beer ({Diapers} → {Beer}).

The Apriori algorithm [6] is an association rule algorithm which is commonly used in business products. This algorithm extracts the frequent item sets from candidate item sets by removing item sets with support values less than minimum support in each iteration. A detailed description of this method would not be presented in this paper, instead only parameters that affect their operation characteristics and performance were described. The reason why this study selected Apriori algorithm for finding association rules as a data mining method was based on the following arguments. Apriori algorithm has been regarded as the most popular and most widely used procedure for finding association between attributes which are related in the given data set. Apriori algorithm is as follows.

Let $I = I_1, I_2, \ldots, I_m$ be a set of m distinct attributes, T be transaction that contains a set of items such that $T \subseteq I$, D be a database with different transaction records. An association rule is an implication in the form of $X \rightarrow Y$, where $X, Y \subset I$ are sets of items called item sets, and $X \cap Y = \phi$. The X is called antecedent while Y is called consequent, the rule means X implies Y.

There are two important basic measures for association rules, named support(S) and confidence(C) [11] [12], which can be defined as follows. The support (S) of an association rule is the ratio of the records that contain $X \cup Y$ to the total number of records in the database, and formulated as follows:

$$\text{Support}\,(X \rightarrow Y, T) = \text{Support}\,(X \cup Y)$$

For a given number of records, confidence (C) is the ratio (in percent) of the number of records that contain $(X \cup Y)$ to the number of records that contain X, and formulated as follows:

$$\text{Conf}\,(X \rightarrow Y, T) = \frac{Supp\,(X \cup Y)}{Supp\,(X)}$$

Step 1: Find all sets of items, which occur with a frequency that is greater than or equal to the user specified threshold support(S).

Step 2: Generate the desired rules using the large item set, which have user-specified threshold confidence(C).
Step 3: Verify the discovered rules using the certainty factor to judge whether the discovered rules are very strong rules or not.

4 Research Framework

There are three steps of the study: (1) Data preprocessing, (2) calculate IC score and (3) association rule mining (Figure1).

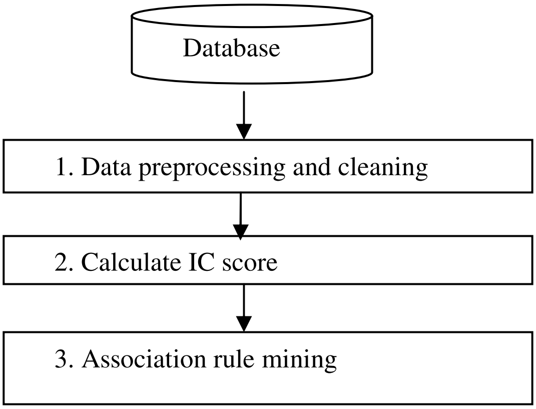

Database
1. Data preprocessing and cleaning
2. Calculate IC score
3. Association rule mining

Fig. 3. Framework of the study

4.1 Data Pre-processing

First step was the cleaning of survey data to remove incomplete, noisy and inconsistent data. Last, data were transformed to be used with statistical software package. In this study, 216 records of survey data set were used for data mining.

4.2 Calculate IC Score

The Intellectual Capital (IC) scores of each organization were calculated by summing all attribute scores obtained from the survey as shown in Table 1. The calculation was done as follows [4]

$$IC = \sum_{k=1}^{n} a_k \tag{1}$$

IC = Intellectual Capital score
a_k = value of each attribute ranged from 0-10

Each organization was classified into class A, B or C by using IC scores. Table 2 presented the classification of Intellectual Capital score. There were 50 organizations in the low class, 133 organizations in the medium and 33 organizations in the high class.

Table 2. Classes of IC scores (n=216)

Class	IC scores
A	Less than 116
B	165-171
C	more than 171

4.3 Association Rule Mining

Apriori algorithm was used for association rule mining algorithm. In the model, 24 features of IC as showned in Table 1 and class of organizations were mined by using Appriori algorithm. Then the relationships of features were proposed in Table 3.

5 Results

After data were divided into 3 classes included low, average and high IC scores. The associations of attributes in each class were analyzed by using Appriori mining algorithm. Measurement indices included Support, Confidence and Lift. Results of association rules of three classes were revealed in Table 3-5.

Table 3. The discovered association rules for Class A (where S, C and L indicate the values of support, confidence and Lift for each discovered rule, respectively.) (support = 0.3 and confidence(C) = 0.86, Lift(L) = 0.8).

id	The discovered association rules	C	L
1	very few employees having training and Class A → very few employees using knowledge in work improvement	0.86	3.0
2	very few employees having training → very few employees using knowledge in work improvement	0.86	3.00
3	very low investments in marketing/revenue and class A→ very few intellectual properties and R&D expenses	1.00	2.33
4	average staff satisfaction and class A → very few intellectual properties	1.00	1.50
5	very low ratio of PC/employee and Class A → very few intellectual properties	0.86	0.85

Class A which had the lowest IC score seemed to be the low performance enterprises. The relationships of their attributes were depicted in Table 3. Results showed that in Class A, very few employees had been trained which resulted in few employees using knowledge to improve the work process. Further the low investment in marketing/revenue and low ratio of PC/employee related to low intellectual properties. In order to increase intellectual properties in this class, the enterprises might increase the investment in marketing, staff satisfaction, and ratio of PC/employee.

Table 4. The discovered association rules for Class B (where S, C and L indicate the values of support, confidence and Lift for each discovered rule, respectively (support = 0.3 and Confidence(C) = 1.0, Lift(L) =1.0)

id	The discovered association rules	C	L
1	very few intellectual properties → Class B	1.0	1.0
2	High customer oriented → Class B	1.0	1.0
3	Average staff competency→ Class B	1.0	1.0
4	High customer satisfaction → Class B	1.0	1.0
5	High numbers of knowledge workers → Class B	1.0	1.0

In Class B, there were some interesting attributes which related to this class. They included very few intellectual properties, high customer oriented, average staff competency, high customer satisfaction and high numbers of knowledge workers. The high value attributes made this Class more IC score than Class A. To increase IC scores in this Class the enterprise should increase staff competency.

Table 5. The discovered association rules for Class C (where S, C and L indicate the values of support, confidence and Lift for each discovered rule, respectively (support = 0.3 and confidence(C) = 0.67, Lift(L) = 1.5)

id	The discovered association rules	C	L
1	Employees with long working years and Class C→ Very low customer lost	0.76	1.64
2	Average IT investment and Class C→ Very high user satisfaction	0.67	1.64
3	Very high experience employees and Class C→ very high knowledge in working	0.67	1.50

Class C which had the highest IC scores, it should be the highest performance enterprise. The relationships of attributes included employees with long working years associated with low customer lost, IT investments implied user satisfaction. The higher employee experiences, the higher knowledge in working was. In order to improve the effectiveness, the enterprise should put more investments in IT.

6 Conclusions

In this study, 216 enterprises in the central part of Thailand were surveyed on their Intellectual Capital. IC scores of each enterprise were calculated from summation of IC attributes. The enterprise was categorized into class by its IC score. Then, association rule mining was applied to the data set. Results showed that enterprises with different Class had different relationships of attributes. Class A which had the low IC scores, having very few intellectual properties. This was related to low investment in marketing, average staff satisfaction, and low ratio of PC/employee. Class B which had average scores showed characteristics of very few intellectual properties as Class A. However, Class B was more customer-oriented which resulted in high customer satisfaction. Class C which was the best performance had many interesting attributes such as staff with very high experiences and long working years.

This characteristic could improve customer loyalty and resulted in very low customer lost. These interesting results should be helpful for strategic management, and planning of Intellectual Capital in organizations.

Acknowledgement

This project was supported by the research grant in year 2007 provided by the Research and Development Institute of Kasetsart University.

References

1. Brooking, A.: Intellectual capital: Current issues and policy implications. J. Intellectual Capital 1(4) (1996)
2. Srivihok, A., Intrapairote, A.: Intellectual Capital measurement: case studies of SMEs in Thailand. In: Proceedings of the International E-Business, Bangkok (2006)
3. Srivihok, A. Intellectual Capital of Enterprises in Thailand: Measurement Model by Bayesian Network Algorithm. In: Proceedings of the tenth International Business Management Association, Kaulalumpur (2008)
4. Chen, J., Zhu, Z., Xie, H.Y.: Measuring Intellectual Capital: a new model and empirical study. J. Intellectual Capital 5(1), 195–212 (2004)
5. Edvinsson, L.: Developing Intellectual Capital at Skandia. Long Range Planning 30(3), 366–373 (1997)
6. Edvinsson, L., Malone, M.S.: Intellectual Capital: Realizing your company's true value by finding its hidden brainpower. Harper - Collins Publishers, New York (1997)
7. Edvinsson, L., Dvir, R., Roth, N., Pasher, E.: Innovations: the new unit of analysis in the knowledge era. The quest and context for innovation efficiency and management of IC, J. Intellectual Capital. 5(1), 40–48 (2004)
8. Gutherie, J.: The management, measurement, and the reporting of intellectual capital. J. Intellectual Capital 2(1), 27–41 (2001)
9. Gutherie, J., Johanson, U., Bukh, P.N., Sanchez, P.: Intangibles and the transparent enterprise: new strands of knowledge. J. Intellectual Capital 4(4), 429–440 (2003)
10. Wiig, K.M.: Integrating Intellectual Capital and Knowledge Management. Long Range Planning 30(3), 399–405 (1997)
11. Agrawal, R., Srikant, R.: Fast algorithms for mining Association Rules. In: Very Large Data Bases, VLDB 20, pp. 487–489. Morgan Kaufmann Publishers Inc., San Francisco (1994)
12. Agrawal, R., Imielinski, T., Swami, A.: Mining association rules between sets of items in large databases. In: Buneman, P., Jajodia, S. (eds.) Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, Washington, DC, pp. 207–216 (1993)
13. WEKA (2009), `http://www.cs.waikato.ac.nz/ml/weka`

Attitudes toward Using Communication Technologies in Education: A Comparative Study of Email and SMS

Boonlert Watjatrakul[1] and Luke Ashim Barikdar[2]

[1] Department of Information Technology, Faculty of Science and Technology,
Assumption University, Bangkok 10240, Thailand
boonlert@scitech.au.edu
[2] Computer Department, Ramkhamhaeng Advent International School,
Bangkok 10240, Thailand
barikdar@gmail.com

Abstract. Educational institutions deploy email and short message service (SMS) to maintain efficient communication with their students. This research examines factors influencing students' attitudes toward using SMS and email, and compares the differences in the proposed factors between email and SMS. The results show that information richness and mobility affect students' perceived utility of email and SMS while information privacy and perceived utility affect the students' attitudes toward using email and SMS. Social pressure has found no impact on the research model. Students also perceive that email provides rich information and utility higher than SMS but SMS possesses mobility more than email. In addition, students have attitudes toward using email more than SMS to maintain communication with their institutions. The paper concludes with a discussion of findings, implications and limitations.

Keywords: Attitudes, Email, SMS, Education, Communication, Technology.

1 Introduction

Communication technologies including email, SMS, instant messaging, and Blog are rapidly becoming inescapable tools for e-business success in a digital era. Electronic mail (email) and Short Message Service (SMS) are important tools that have been long employed to improve organizational performance and support marketing activities in a business sector. Email is rapidly becoming a preferred communication medium for many people. Radicati Group estimates the number of emails sent per day in 2008 is around 210 billion messages and more than 2 million emails are sent every second [7]. Most people routinely use email for work, socialization, and marketing purposes. Email marketing revenue in the U.S. was expected to move up to $ 6.1 billion in 2008 [12]. SMS is a communication service using standardized communications protocols allowing the interchange of short text messages between mobile telephone devices [13]. The number of SMS usages is dramatically increased. Portio Research predicts that SMS will become a US$100 billion by 2010, and worldwide total traffic will reach almost 5 trillion messages in 2011 [14]. SMS traffic in the Asia Pacific region is expected to increase to over 1.2 trillion messages by 2010

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 191–201, 2009.

and revenues earned from SMS usage is estimated to grow to US$15.1 billion [15]. In addition, SMS are being used for marketing purposes including advertising and event participation.

Educational institutions are urged to adopt these technologies to improve communication with their students more efficiently. Most universities supplement or change their traditional communication channels (i.e., telephone calls and letters) to a new way of communication using innovative technologies including email and SMS due to their cost effectiveness, flexibility, immediacy, ubiquity, traceability, and privacy issues [1,2]. Some universities start using email and SMS to communicate with their students in several purposes including requesting a reason for unauthorized absences, sending reminders or changes of calendar dates or appointments, sending reminders to sign up for trips or returning needed forms, informing cancellation of events or emergency for school closing, and informing academic results [8,24]. Recently, the growing body of academic research has focused on examining the determinants of computer technology acceptance (e.g.,[4,5,6]). Little research, however, underlines the students' adoption of communication technologies particularly SMS in education and attempts to understand why students have different preferences to use email and SMS in education.

The primary objectives of this research are to extend the technology adoption in the context of education and examine whether or not antecedents of technology adoption developed with respect to email generalized to SMS. In particular, this research seeks to examine the factors that influence students' attitudes toward using two communication technologies, email and SMS, in the context of education, and compares the differences of the proposed factors between these two communication technologies. The study investigates the effects of technology attributes–information privacy, mobility, information richness, and perceived utility–and social pressure on students' attitudes toward using email and SMS. The finding of this research will help educational institutions to understand why students have different preferences to use email and SMS for communication with their institutions. It also provides guidance for the institutions to use these technologies in education more efficiently.

In the next section, we discuss the proposed factors influencing students' attitudes toward using email and SMS and hypotheses development. This is followed by research design, analysis results, and a discussion of findings, implications and limitations.

2 Factors Influencing Attitudes toward Using Email and SMS

Since the mid 1990s, the adoption of innovative technologies has gained considerable importance as a field of academic research [32]. The success or failure of technology adoption bases on the purpose of usages and users' attitudes toward using the technology. This research focuses on the two technologies–email and SMS–for communication under an education context and examines students' attitudes toward using these technologies based on the technology attributes and social influence; information richness, mobility, information privacy, perceived utility and social pressure.

2.1 Information Richness

Information richness facilitates shared meaning, insight, and understanding within a time interval [18]. All communication media such as telephone, conventional mail, and email possess attributes that lead to distinct richness capacities. Media that foster shared meaning, perceptiveness, and rapid understanding are considered rich. Richer medium contains more types of information and interactivities. It allows users to specify messages for a particular recipient and to have wide-ranging transmission and reception of messages [20]. For instance, video teleconference is richer than a textual internet chat. Rich information enables users to communicate more meaning and better understand ambiguous messages. Students, therefore, consider the media that convey rich information are useful for communication. In other words, if students find that using that medium can present the message more understandable to them, they will consider email or SMS are useful to maintain communication with their universities. Information richness, therefore, have a positive influence on the medium utility perceived by students. In comparison to SMS, email is a richer medium as it can contain more content (e.g., message length and graphic) that decreases equivocal messages.

H1: Information richness will have a positive influence on perceived utility of SMS and email.

H2: Email will provide information richer than SMS.

2.2 Mobility

Mobility is the extent to which the tasks performed by the particular user require him or her to be away from his or her work environment [22,24]. When users have abilities to access and use technologies in anywhere and anytime, they will work more effectively and efficiently that results in more positive perceptions of utility of the technologies. When students stay outside classroom or university, they can efficiently maintain contact with their universities using their computer to access email or using mobile phones to receive SMS. In general, communication technologies offering more mobility are perceived more useful. Messages sent and received via mobile phones yield mobility for students more than messages sent and received via computers.

H3: Mobility will have a positive influence on perceived utilities of SMS and email.

H4: SMS will possess mobility more than email.

2.3 Information Privacy

Information privacy has a potential impact on human interaction in media-based communications [17,23]. It refers to the protection of sensitive and personal information from unintentional and intentional attacks and disclosures [23]. Witmer [27] identified two factors that affect level of privacy: feeling of privacy and system privacy. Feeling of privacy refers to the perception of privacy psychologically, mentally, or conditionally rather than actual security [17]. If a medium is perceived as more public, a sense of less privacy will occur. In other words, if the users perceive that the use of the media does not require involvement of many people during communication, they get a sense of privacy. On the other hand, system privacy refers to the actual security of technologies which concern about the probability that someone may read, or resend a message to or from sender. In this case, the level of privacy is determined by the users' perceptions in relation to the quality of system and device security. Email has log in and password to ensure privacy and privacy of SMS can be protected by activating PIN code of mobile phones [24]. In general, email and SMS are considered having standard security to ensure some levels of system privacy and having feeling of privacy when messages are used for one-to-one communication. Students might not want others to know what messages they send to and receive from their universities. A more private setting results in an increased attitude toward using email and SMS for communication.

H5: Information privacy will have a positive influence on students' attitudes toward using SMS and email.

2.4 Social Pressure

Social pressure refers to the motivations of individuals who believe they should use technologies for positioning themselves in a society [25]. In other words, individuals are motivated to adopt innovative technologies from their personal desires to gain social status [28,29]. The intensity of social pressure may lead students to use email or SMS to communicate with their universities. Students may feel more connected to the medium which is used by his/her peers and others. For example, if students feel the sense that many of their friends use email or SMS, their feeling or understanding may create a sense of social pressure to use email or SMS to keep the connection and maintain communication in their community which leads to an attitude toward using email or SMS. In sum, the media reflecting high intensity of social pressure will result in higher users' attitudes toward using those media.

H6: Social pressure will have a positive influence on students' attitudes toward using SMS and email.

2.5 Perceived Utility

Perceived utility (usefulness) is related to users' attitudes toward using technologies [9,30,31]. Many businesses claim that consumers will accept technology when they believe that using the technology will enhance their productivities. In addition, perceived utility has been confirmed to be the most important factor affecting user acceptance of technologies [10]. Students are more likely to use email or SMS when they perceive that this medium can help them to maintain communication with universities effectively and efficiently. The media that are considered having high utility have high impact on the students' attitudes toward using those media. In comparison to SMS, email has been used for communication between students and universities for a long time because of its abilities to contain more and various content. Email, therefore, is perceived to have higher utility than SMS in terms of maintaining communication between students and universities.

H7: Perceived utility will have a positive influence on students' attitudes toward using SMS and email.

H8: Email will have perceived utility more than SMS.

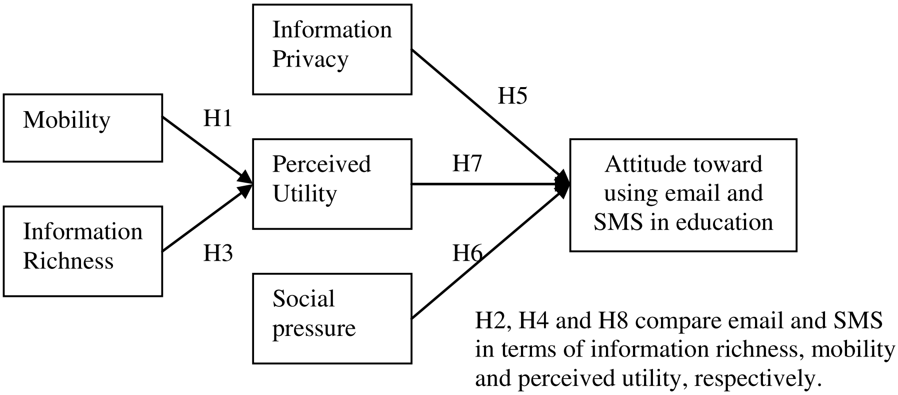

Information Privacy
Mobility
H1
H5
Perceived Utility
H7
Attitude toward using email and SMS in education
Information Richness
H3
H6
Social pressure
H2, H4 and H8 compare email and SMS in terms of information richness, mobility and perceived utility, respectively.

Fig. 1. Proposed Research Model

3 Research Design and Analysis Results

This research employed a survey method as it is an effective approach in gathering data about individual preferences and it can be used to predict individual behaviors [26]. A questionnaire was developed based on the predefined definitions of variables from previous studies (i.e., [5,21,23,24]). It consists of two main parts. The first part contained questions related to email issue while the second part focused on SMS issue. Each question comprised five-point Likert scales (1=strongly disagree, 5=strongly agree). The respondents' demographic questions (e.g., education, nationality and gender) were also included in the questionnaire. To increase validity of the questions, a small group of students randomly selected was asked to complete the questionnaire and then provided comments on any aspect of the questionnaire. The questionnaire was modified based on those comments to improve the clarity of each question.

The modified questionnaire was used to collect data from students in three educational institutions, one international university and two international colleges. To collect data at the university, a brief description of the research topic and requirements were described. The questionnaires then were given to the students and collected back within thirty minutes. The data obtained from two colleges were collected in small groups both inside and outside the classrooms. It took about thirty minutes for each group.

The data were collected from 226 students. Questionnaires having many missing data on both parts–email and SMS–were removed and few missing values were replaced by series means from the statistical analysis software, SPSS. Consequently, 205 complete questionnaires were used for data analysis. The majority of the participants were Thai students (68%) and males (53.2%). 54.6% of the participants were undergraduate students and 45.4% were graduate students. An average age of the participants was 25.6 years.

To check unidimensionality of each scale, a principal component factor analysis with varimax rotation was performed. Items with factor loading values lower than 0.5 were abandoned from further analysis. Table 1 presents the factor loadings and reliability of email and SMS. All items loaded above 0.5 on their respective factors, demonstrating both convergent and discriminant validity [3,11]. All factors had reliabilities of the measurement instruments (Cronbach's alpha,∝) above 0.7, excepting social pressure of email (∝=.65), However, a Cronbach's alpha of 0.6-0.7 indicates acceptable reliability when the removing of any items can not improve the reliability of a factor [16,19].

Table 1. Factor analysis and reliability results

	Email			SMS		
Factors	Items	Cronbach's (∝)	Loadings	Items	Cronbach's (∝)	Loadings
Information Privacy	IPe1	.80	.810	IPs1	.77	.814
	IPe2		.841	IPs2		.852
	IPe3		.743	IPs3		.576
Social Pressure	SPe1	.65	.598	SPs1	.76	.593
	SPe2		.763	SPs2		.820
	SPe3		.811	SPs3		.804
Information Richness	IRe1	.70	.871	IRs1	.75	.688
	IRe2		.810	IRs2		.805
				IRs3		.778
Mobility	MOe1	.73	.869	MOs1	.72	.798
	MOe2		.867	MOs2		.833
				MOs3		.723
Perceived Utility	PUe1	.76	.782	PUs1	.70	.823
	PUe2		.834	PUs2		.685
	PUe3		.693	PUs3		.604
Attitudes	ATe1	.78	.759	ATs1	.78	.685
	ATe2		.876	ATs2		.837
	ATe3		.660	ATs3		.787

A regression analysis was performed to find cause-effect relationships among the respective factors in the research model. Table 2 presents the results of the regression analysis. The results showed that information richness and mobility had statistically significant effects on perceived utility (H1 and H3 were supported). In particular, information richness had a loading value on perceived utility more than mobility for both email and SMS (β=.286>.152; β =.387>.197). In addition, information privacy and perceived utility had statistically significant effects on students' attitudes toward using email and SMS for communication with their universities (H5 and H7 were supported). The results also showed that perceived utility had a loading value on attitudes toward using email and SMS more than information privacy (β=.405>.203; β=.286>.271). Lastly, the results indicated that social pressure had no statistically significant effects on students' attitudes toward using email and SMS (H6 was unsupported).

Table 3 compares the differences between means of the factors in the research model. The results showed that students perceived that email and SMS possessed abilities to provide levels of information richness, mobility, privacy and utility (means > 3). In comparison between email and SMS, the means of information richness, mobility, perceived utility, and attitudes toward use had statistically significant differences. In particular, email had higher information richness than SMS (mean 3.676 > 3.418; H2 was supported). SMS had higher mobility than email (mean 3.874 >3.599; H4 was supported). Email was perceived to have higher utility than SMS (mean 3.954 > 3.720; H8 was supported). Finally, students had attitudes toward using email more than SMS (mean 4.008 > 3.793).

Table 2. Regression analysis results

		Email (beta weights)		SMS (beta weights)	
		Perceived Utility	Attitude	Perceived Utility	Attitude
Information Richness	H1	.286***		.387***	
Mobility	H3	.152*		.197**	
Information Privacy	H5		.203**		.271***
Social Pressure	H6		.115		.096
Perceived Utility	H7		.405***		.286***
r^2		11.4%	32.6%	20.2%	27.6%

*** p<.001; ** p< .01; *p<.05

Table 3. Differences between the factors' means of email and SMS

Factors	Email			SMS			t-test	
	items	Mean	SD.	items	Mean	SD.	t	Sig.(2-tailed)
Information Privacy	3	3.718	.817	3	3.795	.746	1.454	.148
Social Pressure	3	3.724	.756	3	3.711	.862	-0.232	.817
Information Richness	2	3.676	.833	3	3.418	.879	-3.528	.001
Mobility	2	3.599	.921	3	3.874	.732	3.638	.000
Perceived Utility	3	3.954	.728	3	3.720	.751	-4.031	.000
Attitude	3	4.008	.761	3	3.793	.786	-3.884	.000

Table 4 presents the students' attitudes toward using email and SMS in relation to student demographics. The results showed that students in all demographic groups (education, nationality and gender) had an average attitude towards using email and SMS to maintain communication with their institutions (mean>3.0). In particular, graduate students perceived utilities of email and SMS higher than undergraduate students (sig=.009, .041). International students perceived utility and privacy of email higher than Thai students (sig=.006, .021). They also had attitudes towards using email and SMS higher than Thai students (sig=.000, .036). In addition, females had attitudes toward using SMS higher than males (sig=.018).

Table 4. Relationship between demographics and factors affecting the students' attitudes

		items	Information Privacy (mean)		Perceived Utility (mean)		Attitudes toward using (mean)	
			Email	SMS	Email	SMS	Email	SMS
Education	Undergraduate	112	3.7673	3.8418	3.8330	3.6225	3.9464	3.7917
	Graduate	93	3.6584	3.7384	4.1000	3.8370	4.0824	3.7949
	t-test (sig.)		.344	.328	.009**	.041*	.207	.977
Nationality	Thai	139	3.6175	3.7478	3.8583	3.7119	3.8849	3.7165
	International	66	3.9293	3.8939	4.1561	3.7365	4.2677	3.9545
	t-test (sig.)		.021*	.191	.006**	.836	.000**	.036*
Gender	Male	109	3.7446	3.8196	3.9018	3.6314	3.9480	3.6721
	Female	96	3.6875	3.7668	4.0135	3.8202	4.0764	3.9306
	t-test (sig.)		.620	.613	.272	.071	.225	.018*

* p< .05; ** p<.01

4 Discussion of the Findings

The study provides some evidence that students perceive high utilities of email and SMS if they can freely receive, send and check their messages at anywhere (in/outside universities) and anytime (manifestation of mobility). Students perceive that SMS provide more advantage of mobility than email (see Table 3) because students have their mobile phones ready for communication via SMS all the time in contrast to personal computer which is used to access email. The results also show that the utilities of email and SMS are based upon their abilities to increase clarity of the unclear messages (manifestation of information richness). Students perceive that email provide richer information than SMS (see Table 3). This would help them to understand ambiguous messages quicker. SMS, however, might be more appropriate to use for unequivocal message containing few texts [21].

The findings show that students have attitudes toward using email more than SMS to maintain communication with their institutions (university and colleges). The determinants of their attitudes toward using these media are based on their perceptions of utility of the media and privacy of the messages. Students concern that information they send and receive should be private. In other words, the media should maintain confidentiality of communicated information. Email and SMS can deliver private information but provide no statistically significant difference in maintaining private information (see Table 3). Interestingly, perceived utility–based on the media abilities to provide rich information and mobility (state above)–has an impact more than information privacy on students' attitudes toward using both email and SMS (see Table 2). In addition, social pressure does not provide significant effect on students' attitude toward using email and SMS for communication with their institutions. In other words, social environments (e.g., friends, community) do not convince users to use email and SMS in an education circumstance. In this case, students may want to receive specific email and short text messages from their institutions for personal messages rather than the generic messages that everyone can receive like advertising messages.

In addition, the demographic analysis results indicate that international students have attitudes toward using email and SMS to communicate with their institutions more than local or Thai students (Table 4). Perhaps, international students feel more confident to use written communication such as email and SMS and try to avoid conversation trouble as a result of English accent. On the other hand, Local students have more communication channels to contact with the university's staff such as telephone calls and office visits. The results also show that graduate students perceive email and SMS are useful more than undergraduate students (Table 4). Unlike undergraduate students who study full-time, most graduate students are working people and study in the evening or weekend classes. They might prefer to be contacted when they are free or when the contact does not interrupt their works. In this reason, email and SMS seem to be more appropriate and useful for graduate students to maintain communication with their institutions rather than using a phone call during their working time.

5 Implications and Limitations

The study provides some practical implications. Firstly, educational institutions (universities or colleges) may consider using email rather than SMS to communicate with students if the messages need substantial explanation to clarify information and the message is not in an urgent need, such as sending reminders or changes of calendar dates or appointments, and sending reminders to sign up for trips or returning needed forms. Secondly, SMS will be more desirable to be used when the messages are unambiguous and short (i.e., informing academic results in the form of letter grades or GPAs), need urgent recognition within a short time (i.e., informing cancellation of events or emergency closings of the university), or need a fast response by replying messages or calling back. Finally, the institutions should make students feel that email or SMS is useful for them to maintain communication with the universities. For instance, universities should only send message necessary for individual purpose to create feeling of privacy. Students will feel it useful to use email or SMS to contact with their universities and increase potential to read the messages sent by their universities. On the other hand, universities should respond to students' email and text messages rapidly; thus, students can accomplish their tasks faster.

The study has some significant limitations. Firstly, this study addressed five factors affecting the students' attitudes toward using email and SMS in an education context. It did not examine a comprehensive list of factors that might influence on students' acceptance of email and SMS for communication in education. Future research might address other possible drivers of students' attitudes such as costs and risks of using these media. Secondly, this study emphasized the two media, email and SMS, to be used as communication media in education. Other communication technologies such as web board, forum and blog might be useful for the universities to maintain communication with students. Future study might extend this study to investigate other communication technologies. Lastly, the survey method with one-time measurement might call into questions due to inadequate information and the timing of the survey. Future research might reexamine the study hypotheses to validate the study results and improve generalizability of the findings.

6 Conclusions

The study shows evidences that students have attitudes toward using communication technologies, email and SMS, to maintain communication with their institutions (university and colleges). Factors influencing students' attitudes towards using SMS and email are the same including information richness, information privacy, mobility, and perceived utility. However, the effect (loading values) of each factor on students' attitudes toward using email and SMS are statistically significant differences. As opposed to previous studies that confirm the effect of social influence on technology adoption (i.e., [3,25,28]), the study shows that social pressure has no effect on students' attitudes toward using both communication technologies in an education context. Interestingly, this study shows evidences that the factors influencing users' attitudes toward using email can be generalized to SMS and might provide different effects on email and SMS when they are used in different circumstances (i.e., marketing vs. education).

References

1. 10 Reasons to use Email Marketing to beat the Credit Crunch, `http://www.Jasonmillward.com/blog/category/email-marketing`
2. SMS benefits, `http://www.messaging.newmobilemedia.com/sms/sms-benefits.htm`
3. Karahanna, E., Limayem, M.: E-mail and V-mail Usages: Generalizing Across Technologies. Journal of Organizational Computing and Electronic Commerce 10, 49–66 (2000)
4. Matheson, K.: Predicting User Intentions: Comparing the Technology Acceptance Model with the Theory of Planned Behavior. Information Systems Research 2, 173–191 (1991)
5. Davis, F.D.: Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. MIS Quarterly 13, 319–339 (1989)
6. Taylor, S., Todd, P.A.: Understanding Information Technology Usage: A Test of Computing Methods. Information Systems Research 6, 144–176 (1995)
7. How Many Emails Are Sent Every Day?, `http://email.about.com/od/emailtrivia/f/emails_per_day.htm`
8. Watjatrakul, B., Barikdar, L.A.: E-service in Education: The Influences of Media Richness, Social Presence, Privacy and Technology Acceptance Model on Email Adoption. In: the 6th International Conference on e-Business, Bangkok (2007)
9. Hu, P.J., Cheng, Y.K., Sheng, O.L., Tam, K.Y.: Examining the technology acceptance model using physician acceptance of telemedicine. Journal of Management Information Systems 6, 91–112 (1999)
10. Sun, H.: An Integrative Analysis of TAM: Toward a Deeper Understanding of Technology Acceptance Model. In: the 9th Americas Conference on Information Systems, Tampa, Florida (2003)
11. Campbell, D.T., Fiske, D.W.: Convergent and Discriminant Validation by the Multitrait-Multimethod Matrix. Psychological Bulletin 56, 81–105 (1959)
12. Mack, A.M.: E-Mail Marketing Revenue to Triple by 2008, March 18. Adweek (2004)
13. SMS, `http://en.Wikipedia.org/wiki/SMS`
14. Portio Research: Mobile Messaging Futures 2009-2013. Portio Research Ltd (November 2008)

15. Chan, I.: Report: SMS Traffic to Double in AP by 2010. ZDNet Asia (September 08, 2005)
16. Internal Consistency, `http://en.wikipedia.org/wiki/Internal_consistency`
17. Tu, C.H.: The Impacts of Text-based CMC on Online Social Presence. The Journal of Interactive Online Learning 1, 1–24 (2002)
18. Daft, R., Lengel, R., Trevino, L.: Message Equivocally, Media Selection, and Manager Performance: Implications for Information Systems. MIS Quarterly 17, 355–366 (1987)
19. Bullinger, M., Power, M.J., Aaronson, N.K., Cella, D.F., Anderson, R.T.: Creating and Evaluating Cross-Cultural Instruments. In: Spilker, B., Raven, L. (eds.) Quality of Life and Pharmacoeconomics in Clinical Trials, Philadelphia, pp. 659–668 (1996)
20. Dasgupta, S., Granger, M., McGarry, N.: User Acceptance of E-collaboration Technology: An Extension of the Technology Acceptance Model. Group Decision and Negotiation, 87–100 (2002)
21. Bubas, G.: Computer Mediated Communication Theories and Phenomena: Factors that Influence Collaboration over the Internet. In: the 3rd CARNet Users Conference, Zagreb, Croatia (2001)
22. Thanh, D.V., Steensen, S., Audested, J.A.: Mobility Management and Roaming with Mobile Agents. Mobile and Wireless Communications Networks, 123–137 (2000)
23. Weisband, S.P., Reinig, B.A.: Managing User Perceptions of Email Privacy. Communications of the ACM 38, 40–47 (1995)
24. Barikdar, L.A.: Factor Influencing the Utilization of IT Communications between Universities and Students: A Comparative Study of Email versus Short Message Service (SMS). Master's Thesis. Assumption University (2005)
25. Igbaria, M.: User Acceptance of Microcomputer Technology: An Empirical Test. Omega 21, 73–90 (1993)
26. Fowler, F.J.: Survey research methods. Sage, California (2002)
27. Witmer, D.F.: Risky Business: Why People Feel Safe in Sexually Explicit On-line Communication. Journal of Computer Mediated Communication (1997), `http://jcmc.indiana.edu/vol2/issue4/witmer2.html`
28. Kwon, H.S.: A Test of the Technology Acceptance Model: the Case of Cellular Telephone Adoption. In: the 33rd Hawaii International Conference on System Sciences, Maui, Hawaii (2000)
29. Rogers, E.M.: Communication Technology–the New Media in Society. The Free Press, New York (1986)
30. Davis, F.D.: Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. MIS Quarterly 13, 319–339 (1989)
31. Taylor, S., Todd, P.A.: Understanding Information Technology Usage: a Test of Competing Models. Information Systems Research 6, 144–174 (1995)
32. Bauer, H.H., Reichardt, T.S., Bares, J., Neumann, M.M.: Driving Consumer Acceptance of Mobile Marketing: a Theoretical Framework and Empirical Study. Journal of Electronic Commerce Research 6, 181–192 (2005)

Study of Factors Influencing Online Auction Customer Loyalty, Repurchase Intention, and Postitive Word of Mouth: A Case Study of Students from Universities in Taipei, Taiwan

Hsih-Ying Hsu

Graduate School of Business, Assumption University, Bangkok, Thailand
ingrid_hsu@hotmail.com

Abstract. The study aims to find out the factors that keep people repurchasing and remaining loyal to the auction websites. The objectives of the study are to identify the customer loyalty, repurchase intention and positive word of mouth via online auctions in Taiwan. It hopes to study the relationship between e-service quality and e-recovery service quality dimensions toward bidder's disconfirmation. The data was collected from 400 online auction bidders in Taipei, Taiwan. The result shows that there's a direct link between satisfaction and customer's loyalty, repurchase intention, and positive word of mouth. The findings might prove and help the online auction websites to gain a better knowledge of what measures they should take to increase the users' loyalty towards the websites.

Keywords: E-commerce, Repurchase Intention, Customer Loyalty, Positive Word of mouth, Satisfaction.

1 Introduction

In recent years, online auctions represent a large volume of the economic activities over the Internet, and still maintain a high popularity among e-commerce services. There have been millions of auction listings tremendous number of products on auction sites such as eBay, Yahoo and uBid. These business activities have increased rapidly, leading to a retailing revolution – online consumer-to-consumer (C2C) auctions [1]. Online auctions have played a role in many people's lives nowadays. There are a great number of sellers in the online marketplaces for shoppers to choose from. As for the nature of the Internet environment, bidders cannot inspect the real item as they can in traditional stores, which implies they may perceive differences between the imagined and the real product. As a result, auctioneers' (i.e. websites) and sellers' performance will affect buyer's satisfaction and loyalty.

Haeberie [2] stated that one of the main reasons for the popularity of online auctioning is of course the larger available market. Due to the rapid increase in ownership and access to computers and the Internet in the past few years, the online auctioning market has reached a global level which increases the popularity among people.

B. Papasratorn et al. (Eds.): IAIT 2009, CCIS 55, pp. 202–210, 2009.

According to a survey on the use of broadband in Taiwan published by Taiwan Network Information Center (TWNIC), on January 3, 2009, the number Internet users in Taiwan reached 15.8 million with a coverage rate of about 68.94%. Moreover, the survey showed that some 14.2 million Internet users, or nearly 71 percent of the Internet user population in Taiwan, were aged 12 or older. In Carat Media Weekly volume 461 "*A Close Observation On The Status Quo Of On-Line Shopping*", it stated that the chart shows that for the education level of Internet users or Internet shoppers, most people have bachelor degree [3]. Also, it indicated that among the online shoppers who have purchased online, 60% of them had the education level of bachelor degree or higher degree. This is because for this highly educated group of people, Internet has become a necessity in their life and it is more convenient for them to consume online or do the online auction.

This research also emphasizes the expectancy disconfirmation theory (EDT). It is considered as one of the most well-known theories in the marketing and information systems (IS) field, which has gained widespread acceptance in research seeking to explain and predict consumer satisfaction and repurchase intention [4]. This study applied EDT to explore that how e-service quality influences buyer's loyalty, repurchase intention, and positive word of mouth in online auctions in Taiwan.

2 Literature Review

The online auction is unique in the business of e-commerce and has become one of the most successful business innovations on the web [5]. Although online auctions have been popular for some years, the nature of the online environment has implied that C2C transactions are more complex than online shopping. First, one of the most distinguishing features of C2C is that most online parties (i.e. bidders and sellers) usually remain anonymous and transactions are conducted where the relationship between both parties is of an impersonal nature [6] [7]. How to effectively manage those sellers and bidders is the essential issue of the auctioneers. Second, the online auction attracts millions of sellers and anyone can easily become a micro business. A variety of sellers have emerged to deal with the increasing risk associated with online transactions. For example, some of the sellers have traditional stores and are expanding their business through e-commerce; however, most of the others are individuals that participate in auctions for leisure. Thus, the sellers differ greatly in size and quality.

With the popularity of online auctions, bidder behavior related issues have generated topics of great interest. Some previous studies have mostly focused on the auction platform mechanism [8] [9]. It is proposed that customer characteristics and website information content have a positive influence on people's belief in website effectiveness [8]; this, in turn, influences the intention to bid. Some studies examined the relationship of price and bidding behavior [9]-[10]. Some other researchers are interested in understanding the decision process of bidders. It was discovered that the likelihood to bid in online auctions is influenced by access to the computer, ease of use, and involvement with the auction site [11]. Also, it was indicated that starting price, total number of bids, auction duration, and seller's reputation influence the decision dynamics of bidders [12].

This study focused on analyzing the Internet auction characteristics, specifically e-service quality and e-service recovery quality toward online auction affecting customer loyalty and online repurchases intention. The origin of e-service quality as a concept can be traced to the service quality concept. Colier and Bienstock [13] defined e-service quality as “customer's perceptions of the outcome of the service along with recovery perceptions if a problem should occur”. Several studies have attempted to describe the concept by defining the domain of e-service quality and thereby offering a measurement schema.

However, as Hofacker *et al* .[14] observe, efforts at measuring e-service quality have at best led to a modest overlap of dimensions. E-services embody the need satisfaction of traditional services, however by using a new technology. Hence, while components of the traditional service satisfaction models may still retain some meaning, the technology element and the lack of personal contacts in the fulfillment of the service completely transform the customer experience in the context of e-services. Research in the context of technology readiness of consumers [15] and the interaction of consumers with technologically advanced products [16] have established the differences in consumer perspectives while consuming products or services with a strong technology element.

The importance of service quality as an antecedent of customer satisfaction and ultimately customer loyalty has been widely acknowledged [17] [18]. Electronic service quality has previously been defined as “the extent to which a website facilitates efficient and effective shopping, purchasing, and delivery” [19]. This definition appears to be too specific to electronic retailing. In order to capture electronic services in a broader sense, electronic service quality should cover all the offered services and not exclusively transaction-specific elements. In the case of websites intended for informational, promotional or supporting purposes, e-service quality could be defined as “the consumer's evaluation of process and outcome quality of the interaction with a service provider's electronic channels.”

Over the years, there has been an increase in research concerned with post adoption or “continued usage” [20]-[22] rather than the concept of “acceptance” (e.g. technology acceptance model, TAM). One of the most significant results of continued usage research has focused on the repurchase or reuse intention or behavior in IT related usage. Therefore, to help gain a thorough understanding of the underlying phenomena, the Expectancy Disconfirmation Theory (EDT) is presented for the evaluation of continuance usage.

Repeating purchase intention represents the customer’s likelihood of repeatedly purchasing the products or the service in the future [23]. In e-commerce, repurchase behavior of satisfied consumers may be closely related to customer loyalty. Uncles *et al.* [24] argued that customer loyalty can be improved by considering buyer habits and reinforcement in a variety of situations.

Word of mouth represents the customer’s willingness to recommend the product and service to other customers in the future [25] Zeithaml *et al.* [17] suggested that the favorable assessment of service quality leads to favorable behavioral intentions such as positive WOM.

3 Conceptual Framework

This conceptual framework of this study explains the relationship between independent variables and dependent variable. Independent variables consist of e-service quality of the auctioneer and e-recovery service quality of the seller in online auction. For e-service quality of auctioneer, it is measured for efficiency, system availability, and privacy protection. For e-recovery service quality of the seller, it is measured for contact, fulfillment, and responsiveness. The mediators in this framework are comprised of disconfirmation, satisfaction, and attribution. Dependent variables are customer loyalty, repurchase intension, and positive WOM.

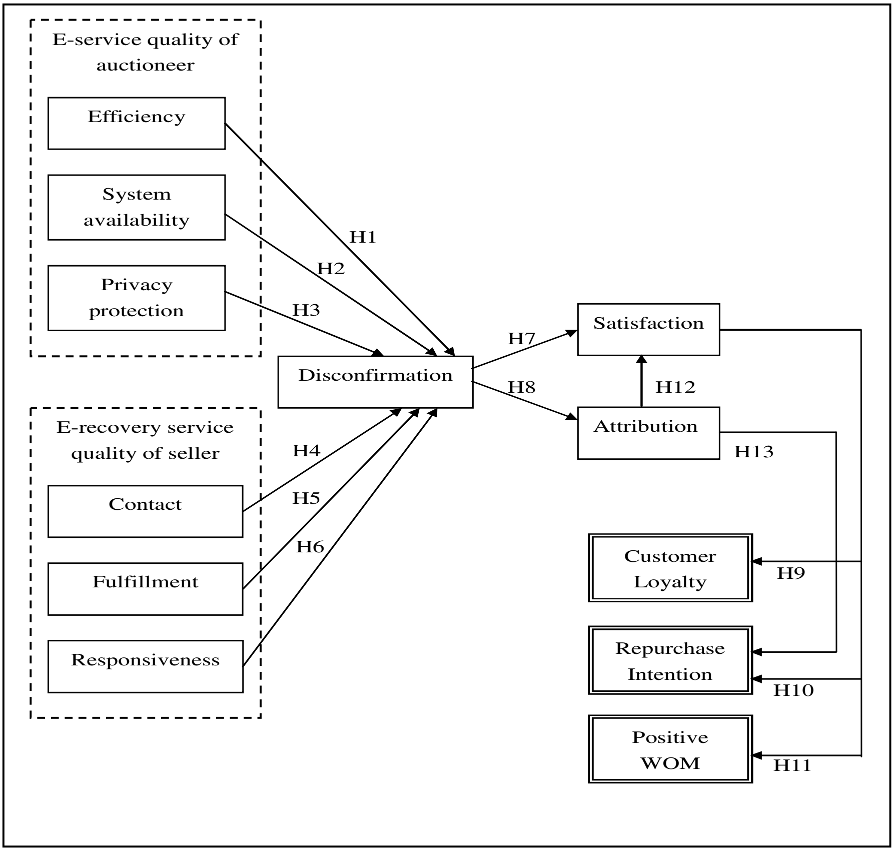


Fig. 1. Conceptual Framework

4 Research Hypothesis

Thirteen hypotheses were developed based on the research objectives and classified into five parts.

Part 1: E-service Quality of the Auctioneer: The relationship of Efficiency, System Availability, and Privacy Protection toward Disconfirmation.

H1o: There is no relationship between efficiency and disconfirmation.

H2o: There is no relationship between system availability and disconfirmation.

H3o: There is no relationship between privacy protection and disconfirmation.

Part 2: E-recovery Service Quality of the Seller: The relationship of Contact, Fulfillment, Responsiveness toward Disconfirmation.

H4o: There is no relationship between contact and disconfirmation.

H5o: There is no relationship between fulfillment and disconfirmation.

H6o: There is no relationship between responsiveness and disconfirmation.

Part 3: Disconfirmation: The Effect of Disconfirmation on Satisfaction and Attribution

H7o: There is no correlation between disconfirmation and satisfaction.

H8o: There is no correlation between disconfirmation and attribution.

Part 4: Satisfaction: The Effect of Satisfaction on Customer Loyalty, Repurchase Intention, Positive WOM

H9o: There is no association between satisfaction and customer loyalty.

H10o: There is no association between satisfaction and repurchase intention.

H11o: There is no association between satisfaction and positive WOM.

Part 5: Attribution: The Effect of Attribution on Satisfaction and on Repurchase Intention

H12o: There is no association between attribution and satisfaction.

H13o: There is no association between attribution and repurchase intention.

5 Research Methodology

In this research, the type of research is considered as descriptive research. In addition, the researchers collect data by distributing questionnaire to respondents; therefore, survey is the research technique. First, the researcher uses *"the stratified sampling method"*, which is deemed as probability sampling. Then the research uses the *"Convenience/Accidental Sampling,"* defined as non-probability sampling.

The analysis of results is based on the data collected from students studying in the national universities in Taipei City. Descriptive analysis and hypothesis testing are two statistical techniques which are used in this analysis as they are the best fit for providing optimal results that can meet the research problems and objectives.

Two sorts of analytical tools have been applied to pave the foundation of the data analysis part in this research;

1. Descriptive Analysis
2. Pearson's Product Moment Correlation Coefficient.

Table 1. Stratified Sampling Method by Using Proportion Technique

Target population (National University Students)	Number of students	Proportion of sampling (percentage)	Proportion of sampling (students)
National Taiwan University (NTU)	31,540	35%	140
National Taiwan Normal University (NTNU)	15,514	17%	68
National Chengchi University (NCCU)	15,377	17%	68
Taipei National University of the Arts (TUNA)	2,065	2%	8
National Taipei University of Education (NTUE)	4,215	5%	20
National Yang-Ming University (NYMU)	4,133	5%	20
National Taiwan University of Science and Technology (NTUST)	10,016	11%	44
National Taipei University of Technology (NTUT)	7,016	8%	32
Total	89,876 students	100%	400 respondents

Table 2. Summary of Hypothesis Testing

Hypothesis	**Statistics Used**	**Significant (Two-tailed) Value**	**Result**
Hypothesis 1	Pearson Correlation Coefficient	0.000	Rejected Ho
Hypothesis 2	Pearson Correlation Coefficient	0.000	Rejected Ho
Hypothesis 3	Pearson Correlation Coefficient	0.000	Rejected Ho
Hypothesis 4	Pearson Correlation Coefficient	0.000	Rejected Ho
Hypothesis 5	Pearson Correlation Coefficient	0.000	Rejected Ho
Hypothesis 6	Pearson Correlation Coefficient	0.000	Rejected Ho
Hypothesis 7	Pearson Correlation Coefficient	0.000	Rejected Ho
Hypothesis 8	Pearson Correlation Coefficient	0.000	Rejected Ho
Hypothesis 9	Pearson Correlation Coefficient	0.000	Rejected Ho
Hypothesis 10	Pearson Correlation Coefficient	0.000	Rejected Ho
Hypothesis 11	Pearson Correlation Coefficient	0.000	Rejected Ho
Hypothesis 12	Pearson Correlation Coefficient	0.000	Rejected Ho
Hypothesis 13	Pearson Correlation Coefficient	0.000	Rejected Ho

5.1 Research Instruments / Questionnaire

The questionnaire has two sections. The first sections is the prescreening question. The second section contains the research questions. After the data were collected, it was transformed and coded using the Statistical Package for the Social Sciences (SPSS).

6 Results and Conclusion

For the demographic factors of the research, the result can be implied as that if people want to sell something through online auction, 90% of them may consider to use Yahoo!Kimo, which is the most widely used platform. Also, the result indicated that most of the people who use online auction to purchase goods are young female. Yet most of the respondents have used online bidding for quite many times, this prove that Internet and online shopping indeed have played an important role in Taiwanese universities students.

As for the hypothesis testing part, from hypothesis one to six, this study confirms that efficiency, privacy protection, contact, fulfillment, and responsiveness have statistically significant influences on disconfirmation. The findings imply that efficiency and system availability of the auctioneer are the most important drivers of the buyer's disconfirmation. Moreover, from hypothesis seven, the findings show that buyers' disconfirmation is positively associated with their satisfaction, and their satisfaction is positively associated with loyalty intention. In other words, bidders will repurchase the goods at the online auction platform that they are satisfied with or through the online auction sellers that generate buyers’ satisfaction.

7 Recommendations

From the auctioneer's perspective, the findings emphasize the importance of the e-service quality of auctioneer. While the previous literature has predominantly focused on e-service quality, this study aims to capture the most important aspects of the auctioneer, particularly in efficiency and system availability. Auctioneer should improve the e-service quality to attract sellers to join this marketplace, and then have competitive advantage from other auctioneers. For instance, they can put more emphasis on the privacy protection of the auction website to assure their customers will not suffer the risks of privacy invasion.

Over the years, there has been tremendous growth of online auction activities, yet online trading is still facing problems. Such problems as missing contact information, inaccurate order delivery, and incomplete after-sales service may lead to a barrier between sellers and buyers. Reputation systems (i.e. rating score, feedback forum) list previous trading activities of sellers and buyers, which also illustrate that every dealing performance could affect the buyer's decision. It is important to note that the online auction websites should create a fair and humane reputation system for the buyers to use after they bid.

Next, the result also indicated that from the buyer's perspective, the strength of the relationship between satisfaction and positive word of mouth has been found to vary significantly. How to create WOM and have a high retention rate becomes the most important issue in the online auction marketplace. Both auctioneers and sellers should especially put more efforts on e-service quality, which are strong enablers for generating future customer cash flows.

Last, the role of attributions in affecting satisfaction and repurchase intentions indicates that seller and auction websites need to pay special attention to policies and practices designed to ensure that customers would have positive purchase and customer service experiences at the website.

8 Limitations and Future Studies

This study is aimed at providing a more practical view of online auction management; however, it still has some limitations. First, the research result can only represent the respondents' ideas in the time period of May, 2009. Another limitation is that This research only discusses about the most important factors of the auctioneer and seller dimensions; other related constructs in those dimensions that may influence disconfirmation should also be examined. Some moderator effect could be tested in the future study, such as the bidder's demographic profile, bidding experience, and product involvement. Further, economic dimensions may add factors such as economy crisis, time and effort, and price premium. Finally, compensation and recovery service show that there is considerable benefit in forming post-purchase satisfaction..

There have been claims that the reputation systems of auction sites such as e-bay have been errors and may not be relied upon. Some challenges still exist in reputation systems [26]. Further research is needed in this area, particularly concerning how to establish a reputation for behaving honestly. On the other hand, the future study can also be about how the reputation system affects the online bidding behavior or how it influence the buyers' repurchase rate or loyalty.

References

[1] Stern, B.B., Stafford, M.R.: Individual and social determinants of winning bids in online auctions. Journal of Consumer Behaviour 5, 43–55 (2006)

[2] Haeberie, M.: Ebay Simplified. Chain Store Age 80(5), 118–199 (2004)

[3] A Close Observation On The Status Quo Of On-Line Shopping. Carat Media Weekly 461, 18–20 (2008)

[4] Yen, C.H., Lu, H.P.: Effects of e-service quality on loyalty intention: an empirical study in online auction. Managing Service Quality 18(2), 127–146 (2008)

[5] Hayne, S.C., Smith, C.A.P., Vijayasarathy, L.R.: Who wins on eBay An analysis of bidders and their bid behaviours. Electronic Markets 13(4), 282–293 (2003)

[6] Kim, Y.: The effects of buyer and product traits with seller reputation on price premiums in e-auction. Journal of Computer Information Systems 46(1), 79–91 (2005)

[7] Resnick, P., Zeckhauser, R.: Trust among strangers in internet transactions: empirical analysis of eBay's reputation system. In: Baye, M.R. (ed.) The Economics of the Internet and E-Commerce. Advances in Applied Microeconomics, vol. 11. Elsevier Science, Amsterdam (2002)

[8] Kwon, O.B., Kim, O., Lee, E.J.: Impact of website information design factors on consumer ratings of web-based auction sites. Behaviour and Information Technology 21(6), 387–402 (2002)

[9] Pavlou, P.A.: Trustworthiness as a source of competitive advantage in online auction markets. In: Best Paper Proceedings, Academy of Management, pp. 1–6. Denver, CO (2002)

[10] Ottaway, T.A., Bruneau, C.L., Evans, G.E.: The impact of auction item image and buyer/seller feedback rating on electronic auctions. Journal of Computer Information Systems 43(3), 56–60 (2003)

[11] Stafford, M.R., Stern, B.: Consumer bidding behavior on internet auction sites. International Journal of Electronic Commerce 7(1), 135–150 (2002)

[12] Ariely, D., Simonson, I.: Buying, bidding, playing, or competing? Value assessment and decision dynamics in online auctions. Journal of Consumer Psychology 13, 113–123 (2003)

[13] Collier, J.E., Bienstock, C.C.: Measuring service quality in e-retailing. Journal of Service Research 8(3), 260–275 (2006)

[14] Hofacker, C.F., Goldsmith, R.E., Bridges, E., Swilley, E.: E-services: a synthesis and research agenda. Journal of Value Chain Management (2007)

[15] Parasuraman, A.: Technological readiness index (TRI): a multiple item scale to measure readiness to embrace new technologies. Journal of Services Research 2(4), 307–320 (2000)

[16] Mick, D.G., Fournier, S.: Paradoxes of technology: consumer cognizance, emotions and coping strategies. Journal of Consumer Research 25, 123–147 (1998)

[17] Zeithmal, V.A., Berry, L.L., Parasuraman, A.: The behavioral consequences of service quality. Journal of Marketing 60, 31–46 (1996)

[18] Anderson, R.E., Fornell, C.: Cross-category variation in customer satisfaction and retention. Marketing Letters 5, 19–30 (1994)

[19] Zeithmal, V.A., Parasuraman, A., Malhotra, A.: A conceptual framework for understanding e-service quality: implications for future research and managerial practice. In: Marketing Science Institute, Cambridge, MA (2000) (working paper)

[20] Bhattacherjee, A., Premkumar, G.: Understanding changes in belief and attitude toward information technology usage: a theoretical model and longitudinal test. MIS Quarterly 28(2), 229–254 (2004)

[21] Ahuja, M.K., Thatcher, J.B.: Moving beyond intentions and toward the theory of trying: effects of work environment and gender on post-adoption information technology use. MIS Quarterly 29(3), 427–459 (2005)

[22] Hsu, M.H., Yen, C.H., Chiu, C.M., Chang, C.M.: A longitudinal investigation of continued online shopping behavior: an extension the theory of planned behavior. International Journal of Human-Computer Studies 64(9), 889–904 (2006)

[23] Parasuraman, A., Grewal, D.: The Impact of Technology on the Quality-Value-Loyalty Chain: A Research Agenda. Journal of the Academy of Marketing Science, 168–174 (2000)

[24] Uncles, M.D., Dowling, G.R., Hammond, K., Manaresi, A.: Consumer loyalty marketing in repeat-purchase markets. In: pp. 98–202. London Business School, London (1998)

[25] Dabholkar, P., Thorpe, D.I., Rents, J.Q.: A measure of service quality for retail stores. Journal of the Academy of Marketing Science 24(1), 3–16 (1995)

Author Index